

IN THE UNITED STATES DISTRICT COURT  
FOR THE DISTRICT OF DELAWARE

CIRBA INC. (d/b/a DENSIFY),  
and CIRBA IP, INC.,

*Plaintiffs/Counter-Defendants,*

v.

VMWARE, INC.,

*Defendant/Counter-Plaintiff.*

C.A. No. 19-00742-LPS

**JOINT CLAIM CONSTRUCTION BRIEF**

OF COUNSEL:

Arturo J. González  
Michael A. Jacobs  
Richard S. J. Hung  
MORRISON & FOERSTER LLP  
425 Market Street  
San Francisco, CA 94105  
(415) 268-7000  
agonzalez@mofo.com  
mjacobs@mofo.com  
rhung@mofo.com

Bitra Rahebi  
MORRISON & FOERSTER LLP  
707 Wilshire Boulevard  
Los Angeles, CA 90017  
(213) 892-5200  
brahebi@mofo.com

Scott F. Llewellyn  
MORRISON & FOERSTER LLP  
4200 Republic Plaza  
370 Seventeenth Street  
Denver, CO 80202-5638  
(303) 592-2204  
sllewellyn@mofo.com

YOUNG CONAWAY STARGATT &  
TAYLOR, LLP

Anne Shea Gaza (No. 4093)  
Robert M. Vrana (No. 5666)  
Samantha G. Wilson (No. 5816)  
Rodney Square  
1000 North King Street  
Wilmington, DE 19801  
(302) 571-6600  
agaza@ycst.com  
rvrana@ycst.com  
swilson@ycst.com

**ATTORNEYS FOR COUNTER-  
PLAINTIFF VMWARE, INC.**

Sarah O. Jorgensen (*pro hac vice*)  
sjorgensen@reichmanjorgensen.com  
REICHMAN JORGENSEN LLP  
1201 West Peachtree, Suite 2300  
Atlanta, GA 30309  
Telephone: (404) 609-1040  
Telecopier: (650) 623-1449

Christine E. Lehman (*pro hac vice*)  
clehman@reichmanjorgensen.com  
Adam Adler (*pro hac vice*)  
aadler@reichmanjorgensen.com  
REICHMAN JORGENSEN LLP  
1615 M Street, NW, Suite 300  
Washington, DC 20036  
Telephone: (202) 894-7310  
Telecopier: (650) 623-1449

Khue V. Hoang (*pro hac vice*)  
khoang@reichmanjorgensen.com  
Jaime Cardenas-Navia (*pro hac vice*)  
jcardenas-navia@reichmanjorgensen.com  
Wesley L. White (*pro hac vice*)  
wwhite@reichmanjorgensen.com  
Rahul Sarkar (*pro hac vice*)  
rsarkar@reichmanjorgensen.com  
REICHMAN JORGENSEN LLP  
750 Third Avenue, Suite 2400  
New York, NY 10017  
Telephone: (212) 381-1965  
Telecopier: (650) 623-1449

Kenneth L. Dorsney (#3726)  
kdorsney@morrisjames.com  
MORRIS JAMES LLP  
500 Delaware Avenue, Suite 1500  
Wilmington, DE 19801  
Telephone: (302) 888-6800

Courtland L. Reichman (*pro hac vice*)  
creichman@reichmanjorgensen.com  
Shawna Ballard (*pro hac vice*)  
sballard@reichmanjorgensen.com  
Jennifer P. Estremera (*pro hac vice*)  
jestremera@reichmanjorgensen.com  
Michael G. Flanigan (*pro hac vice*)  
mflanigan@reichmanjorgensen.com  
Joachim B. Steinberg (*pro hac vice*)  
jsteinberg@reichmanjorgensen.com  
Kate Falkenstien (*pro hac vice*)  
kfalkenstien@reichmanjorgensen.com  
Ariel C. Green (*pro hac vice*)  
agreen@reichmanjorgensen.com  
REICHMAN JORGENSEN LLP  
100 Marine Parkway, Suite 300  
Redwood Shores, California 94065  
Telephone: (650) 623-1401  
Telecopier: (650) 623-1449

Gary J. Toman (*pro hac vice*)  
Gtoman@wwhgd.com  
WEINBERG WHEELER HUDGINS GUNN & DIAL  
3344 Peachtree Road NE, Suite 2400  
Atlanta, GA 30326  
Telephone: (404) 876-2700  
Telecopier: (404) 875-9433

Peter J. Ayers (*pro hac vice*)  
peter@ayersiplaw.com  
LAW OFFICE OF PETER J. AYERS, PLLC  
2200 Bowman Avenue  
Austin, TX 78703  
Telephone: (512) 771-3070  
Telecopier: (512) 520-4459

**ATTORNEYS FOR COUNTER-  
DEFENDANTS**

**DENSIFY’S TABLE OF ABBREVIATIONS**

<b>Abbreviation</b>	<b>Document(s)</b>
'049	U.S. Patent No. 8,336,049
'049 FH	File history for Appl. No. 12/366,008 (now '049)
'266	U.S. Patent No. 8,875,266
'266 FH	File history for Appl. No. 12/122,582 (now '266)
'151	U.S. Patent No. 9,521,151
'151 FH	File history for Appl. No. 14/520,727 (now '151)
'752	U.S. Patent No. 10,069,752
'752 FH	File history for Appl. No. 15/056,027 (now '752)
Madisetti Dec.	Declaration of Dr. Vijay K. Madisetti (Jun. 9, 2020)
POSA	Person of ordinary skill in the art

## TABLE OF CONTENTS

	Page
I. AGREED-UPON CONSTRUCTIONS .....	1
A. U.S. Patent No. 8,336,049 (“the ’049 patent”) .....	1
1. generate .....	1
B. U.S. Patent No. 8,875,266 (the “’266 patent”) .....	1
1. Preamble (Claim 1) .....	1
2. “hardware platform” (Claim 1) .....	1
3. “host platform” (Claim 1) .....	1
C. U.S. Patent No. 9,521,151 (the “’151 patent”) .....	1
1. Preamble (Claims 1 and 12) .....	1
D. U.S. Patent No. 10,069,752 (the “’752 patent”) .....	2
1. Preamble (Claims 1 and 9) .....	2
2. “snapshots” (Claims 1 and 9) .....	2
II. DENSIFY’S LEGAL PRINCIPLES OF CLAIM CONSTRUCTION .....	2
III. DENSIFY’S LEVEL OF ORDINARY SKILL IN THE ART .....	4
IV. DISPUTED CONSTRUCTIONS .....	4
A. U.S. Patent No. 8,336,049 (“the ’049 patent”) .....	4
1. “provisioning [virtual machines]” (Claims 1, 10, and 16) .....	4
2. “virtual machine utilization for the application” (Claims 1, 10, and 16) .....	11
3. “receiving/received resource utilization statistics” (Claims 1, 8–10, 13, 14, 16, 19, and 20) .....	15
4. “transmit[ting] instructions relating to provisioning virtual machines” (Claims 1, 10, and 16) .....	17
B. U.S. Patent No. 8,875,266 (“the ’266 Patent”) .....	21
1. “identifying” (Claim 1) .....	21
2. “prohibiting the virtual machine from executing” (Claim 1) .....	24
3. “permitting the virtual machine to execute” / “authorized to execute” (Claim 1) .....	26
4. “policies” (Claim 1) .....	27
C. U.S. Patent No. 9,521,151 (“the ’151 Patent”) .....	30
1. “optimizing” / “optimize” (Independent Claims 1, 12) .....	30

**TABLE OF CONTENTS**  
(continued)

		<b>Page</b>
2.	“operating conditions” (Independent Claims 1, 12) .....	34
3.	“the at least one policy set” (Independent Claims 1, 12).....	37
4.	“objective” (Dependent Claims 3, 4, 14, 15) .....	40
5.	“the desired state” (Independent Claims 1, 12) .....	42
6.	“user-defined threshold” (Dependent Claims 2, 13).....	45
7.	“at least one recommended action” (Dependent Claims 2, 5, 6) .....	48
8.	“a recommended change to the operating conditions of the cloud environment to achieve at least one desired state consistent with the at least one user-defined policy” (Independent Claims 1, 12) .....	50
D.	U.S. Patent No. 10,069,752 (“752 Patent”) .....	53
1.	“interfacing with the distributed computer systems at a remote resource allocation module that is located outside of the distributed computer systems” (Claims 1, 9) .....	53
2.	“remote resource allocation module” (Claims 1, 9).....	59
3.	“receiving a plurality of computer system snapshots” (Claims 1, 9).....	70
4.	“as defined by the computer system snapshots” (Claims 3, 11) .....	75

**TABLE OF AUTHORITIES****Page(s)****Cases**

<i>3G Licensing, S.A. v. Blackberry Ltd.</i> , No. 17-82-LPS, 2018 WL 4375091 (D. Del. Sept. 13, 2018) .....	18
<i>Acumed LLC v. Stryker Corp.</i> , 483 F.3d 800 (Fed. Cir. 2007).....	4, 12
<i>Affinity Labs of Tex., LLC v. Clear Channel Broad., Inc.</i> , No. 12-205, 2014 WL 1699063 (W.D. Tex. Apr. 29, 2014) .....	4
<i>Alarm.com, Inc. v. SecureNet Techs. LLC</i> , 345 F. Supp. 3d 544 (D. Del. 2018).....	45
<i>Apple, Inc. v. Ameranth, Inc.</i> , 842 F.3d 1229 (Fed. Cir. 2016).....	26
<i>Arendi S.A.R.L. v. LG Electronics, Inc.</i> , No. 12-1595-LPS, 2019 WL 3891150 (D. Del. Aug. 19, 2019).....	<i>passim</i>
<i>Aristocrat Techs. Australia Pty Ltd. v. Int'l Game Tech.</i> , 521 F. 3d 1328 (Fed. Cir. 2008).....	63
<i>Avenue Innovations, Inc. v. E. Mishan &amp; Sons Inc.</i> , 310 F. Supp. 3d 457 (S.D.N.Y. 2018).....	19
<i>BASF Corp. v. Johnson Matthey Inc.</i> , 875 F.3d 1360 (Fed. Cir. 2017).....	45
<i>Catalina Mktg. Int'l, Inc. v. Coolsavings.com, Inc.</i> , 289 F.3d 801 (Fed. Cir. 2002).....	31
<i>Chrimar Holding Co., LLC v. ALE USA Inc.</i> , 732 F. App'x 876 (Fed. Cir. 2018), <i>as amended</i> .....	64, 65
<i>Cox Commc'ns, Inc. v. Sprint Commc'ns Co.</i> , 838 F.3d 1224 (Fed. Cir. 2016).....	51
<i>Creo Prods., Inc. v. Presstek, Inc.</i> , 305 F.3d 1337 (Fed. Cir. 2002).....	60
<i>CSS Tech., Inc. v. Panduit Corp.</i> , 778 F. App'x 947 (Fed. Cir. 2019) .....	54, 71
<i>Curtiss-Wright Flow Control Corp. v. Velan, Inc.</i> , 438 F.3d 1374 (Fed. Cir. 2006).....	6

**TABLE OF AUTHORITIES**  
(continued)

	<b>Page</b>
<b>Cases</b>	
<i>Dickinson &amp; Co. v. C.R. Bard Inc.</i> , 922 F.2d 792 (Fed. Cir. 1990).....	36, 54
<i>Edwards Lifesciences LLC v. Cook Inc.</i> , 582 F.3d 1322 (Fed. Cir. 2009).....	6, 7
<i>Elm 3DS Innovations, LLC v. Samsung Elecs. Co.</i> , 2020 WL 1850657 (D. Del. Apr. 13, 2020).....	19
<i>H. Lundbeck A/S v. Apotex Inc.</i> , No. CV 18-88-LPS, 2019 WL 3206016 (D. Del. July 16, 2019) .....	8
<i>Helmsderfer v. Bobrick Washroom Equip., Inc.</i> , 527 F.3d 1379 (Fed. Cir. 2008).....	39
<i>Hill-Rom Servs., Inc. v. Stryker Corp.</i> , 755 F.3d 1367 (Fed. Cir. 2014).....	23, 24
<i>Indacon, Inc. v. Facebook, Inc.</i> , 824 F.3d 1352 (Fed. Cir. 2016).....	4, 5, 7
<i>Interval Licensing LLC v. AOL, Inc.</i> , 766 F.3d 1364 (Fed. Cir. 2014).....	7, 53, 55
<i>IOEngine, LLC v. Interactive Media Corp.</i> , No. 14-1571-GMS, 2016 WL 1121938 (D. Del. Mar. 21, 2016) .....	20
<i>Jazz Pharm., Inc. v. Amneal Pharm., LLC</i> , 895 F.3d 1347 (Fed. Cir. 2018).....	21, 22
<i>Kaneka Corp. v. Xiamen Kingdomway Grp. Co.</i> , 790 F.3d 1298 (Fed. Cir. 2015).....	71, 72
<i>Kyocera Wireless Corp v. ITC</i> , 545 F.3d 1340 (Fed. Cir. 2008).....	45
<i>Liebel-Flarsheim Co. v. Medrad, Inc.</i> , 358 F.3d 898 (Fed. Cir. 2004 (citation omitted)).....	4, 23
<i>M2M Sols. LLC v. Sierra Wireless Am., Inc.</i> , No. 12-30-RGA, 2015 WL 5826816 (D. Del. Oct. 2, 2015) .....	57
<i>M2M Sols. LLC v. Sierra Wireless Am., Inc.</i> , No. 12-30-RGA, 2016 WL 1298961 .....	57, 62

**TABLE OF AUTHORITIES**  
(continued)

	<b>Page</b>
<b>Cases</b>	
<i>Magna Elecs., Inc. v. TRW Auto. Holdings Corp.</i> , No. 12-654, 2015 WL 11401855 (W.D. Mich. Apr. 28, 2015).....	34, 35, 45
<i>Mantech Envtl. Corp. v. Hudson Envtl. Servs., Inc.</i> , 152 F.3d 1368 (Fed. Cir. 1998).....	42, 43
<i>Mas-Hamilton Grp. v. LaGard, Inc.</i> , 156 F.3d 1206 (Fed. Cir. 1998).....	67
<i>Microprocessor Enhancement Corp. v. Tex. Instruments Inc.</i> , 520 F.3d 1367 (Fed. Cir. 2008).....	18, 19
<i>Mycone Dental Supply Co. v. Creative Nail Design, Inc.</i> , No. 12-00747 RS, 2013 WL 3656379 (N.D. Cal. July 12, 2013).....	34
<i>Novo Indus., L.P. v. Micro Molds Corp.</i> , 350 F.3d 1348 (Fed. Cir. 2003).....	37
<i>Omega Eng'g, Inc. v. Raytek Corp.</i> , 334 F.3d 1314 (Fed. Cir. 2003).....	8
<i>Pacing Techs., LLC v. Garmin Int'l, Inc.</i> , 778 F.3d 1021 (Fed. Cir. 2015).....	56
<i>Personalized Media Comm'ns, LLC v. ITC</i> , 161 F.3d 696 (Fed. Cir. 1998).....	59, 66
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005).....	passim
<i>Red Rock Analytics, LLC v. Samsung Elecs. Co., Ltd.</i> , 2018 WL 1806859 (E.D. Tex. Apr. 16, 2018).....	69
<i>In re Skvorecz</i> , 580 F.3d 1262 (Fed. Cir. 2009).....	34, 35
<i>Sonix Tech. Co. v. Publ'ns Int'l, Ltd.</i> , 844 F.3d 1370 (Fed. Cir. 2017).....	18
<i>Sound View Innovations, LLC v. Facebook, Inc.</i> , No. 16-116, 2017 WL 2221177 (D. Del. May 19, 2017) .....	65, 66
<i>Texas Instruments Inc. v. United States Int'l Trade Comm'n</i> , 988 F.2d 1165 (Fed. Cir. 1993).....	36, 54



**TABLE OF AUTHORITIES**  
(continued)

	<b>Page</b>
 <b>Cases</b>	
<i>Toshiba Corp. v. Imation Corp.</i> , 681 F.3d 1358 (Fed. Cir. 2012).....	23, 39
<i>TQ Delta, LLC v. 2Wire, Inc.</i> , 2018 WL 626472 (D. Del. Jan. 30, 2018).....	61
<i>U.S. Auto. Ass’n v. Wells Fargo Bank, N.A.</i> , 2019 WL 2488135 (E.D. Tex. June 13, 2019).....	9
<i>Uniloc USA, Inc. v. Microsoft Corp.</i> , 290 F. App’x 337 (Fed. Cir. 2008) .....	4
<i>Viatech Techs., Inc. v. Microsoft Corp.</i> , C.A. No. 14-1226-RGA, 2016 WL 3398025 (D. Del. Jun. 14, 2016).....	66
<i>Vitronics Corp. v. Conceptronic, Inc.</i> , 90 F.3d 1576 (Fed. Cir. 1996).....	4
<i>W.E. Hall Co., Inc. v. Atlanta Corrugating, LLC</i> , 370 F.3d 1343 (Fed. Cir. 2004).....	9
<i>Williamson v. Citrix Online, LLC</i> , 792 F.3d 1339 (Fed. Cir. 2015).....	60, 61, 62, 65
<i>Yodlee, Inc. v. Plaid Techs., Inc.</i> , No. 14-1445-LPS, 2016 WL 204372 (D. Del. Jan. 15, 2016) .....	18, 19
<i>Zeroclick, LLC v. Apple Inc.</i> , 891 F.3d 1003 (Fed. Cir. 2018).....	57, 60, 61, 65
 <b>Statutes</b>	
35 U.S.C. § 112(f).....	<i>passim</i>
 <b>Other Authorities</b>	
M.P.E.P § 2181 .....	61

**I. AGREED-UPON CONSTRUCTIONS****A. U.S. Patent No. 8,336,049 (“the ’049 patent”)****1. “generate . . . receive . . . adjust . . . transmit” (Claims 1, 10, and 16)**

<b>Claim Term/Phrase</b>	<b>Agreed-Upon Proposed Construction</b>
<b><u>Claim 1, 10, and 16</u></b>  “generate . . . receive . . . adjust . . . transmit”	The parties agree that the claim steps are ordered/sequenced.

**B. U.S. Patent No. 8,875,266 (the “’266 patent”)****1. Preamble (Claim 1)**

<b>Claim Term/Phrase</b>	<b>Agreed-Upon Proposed Construction</b>
<b><u>Claim 1</u></b>  Preamble	The parties agree that the preamble to claim 1 of the ’266 patent is limiting.

**2. “hardware platform” (Claim 1)**

<b>Claim Term/Phrase</b>	<b>Agreed-Upon Proposed Construction</b>
<b><u>Claim 1</u></b>  “hardware platform”	“physical system hardware”

**3. “host platform” (Claim 1)**

<b>Claim Term/Phrase</b>	<b>Agreed-Upon Proposed Construction</b>
<b><u>Claim 1</u></b>  “host platform”	“physical system hardware running virtualization software”

**C. U.S. Patent No. 9,521,151 (the “’151 patent”)****1. Preamble (Claims 1 and 12)**

<b>Claim Term/Phrase</b>	<b>Agreed-Upon Proposed Construction</b>
<b><u>Claim 1, 12</u></b>  Preamble	The parties agree that the preambles to claims 1 and 12 of the ’151 patent are limiting.

**D. U.S. Patent No. 10,069,752 (the “’752 patent”)****1. Preamble (Claims 1 and 9)**

<b>Claim Term/Phrase</b>	<b>Agreed–Upon Proposed Construction</b>
<b><u>Claim 1, 9</u></b> Preamble	The parties agree that the preambles to claims 1 and 9 of the ’752 patent are limiting.

**2. “snapshots” (Claims 1 and 9)**

<b>Claim Term/Phrase</b>	<b>Agreed–Upon Proposed Construction</b>
<b><u>Claims 1, 9</u></b> “snapshots”	“data that contains configuration and resource usage information of a distributed computer system at a particular moment in time”

**II. DENSIFY’S LEGAL PRINCIPLES OF CLAIM CONSTRUCTION**

The patent “specification is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.” *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996). “While ‘the claims themselves provide substantial guidance as to the meaning of particular claim terms,’ the context of the surrounding words of the claim also must be considered. *Arendi S.A.R.L. v. LG Electronics, Inc.*, 2019 WL 3891150, \*2 (D. Del. Aug. 8, 2019) (J. Stark) (quoting *Phillips v. AWH Corp.*, 415 F.3d 1303, 1314 (Fed. Cir. 2005)). A “specification may reveal a special definition given to a claim term by the patentee that differs from the meaning it would otherwise possess. In such cases, the inventor’s lexicography governs.” *Phillips*, 415 F.3d at 1316. A court may consider extrinsic evidence, but it is generally considered “less reliable” than intrinsic evidence. *Id.* at 1318-19. “Where the intrinsic record unambiguously describes the scope of the patented invention, reliance on any extrinsic evidence is improper. *Arendi*, 2019 WL 3891150 at \*2; *see also Pitney Bowes, Inc. v. Hewlett-Packard Co.*, 182 F.3d 1298, 1308 (Fed. Cir. 1999).

A patent claim is indefinite if, viewed in light of the specification and prosecution history, it fails to inform those skilled in the art about the scope of the invention with reasonable certainty.”

*Elm 3DS Innovations, LLC v. Samsung Elecs. Co.*, No. 1401439, 2020 WL 1850657, at \*3 (D. Del. Apr. 13, 2020) (quoting *Nautilus, Inc. v. Biosig Instruments, Inc.*, 572 U.S. 898, 909-10). “A patent does not satisfy the definiteness requirement of § 112 merely because a court can ascribe *some* meaning to a patent’s claims.” *IBM v. Priceline Grp. Inc.*, 271 F. Supp. 3d 667, 678 (quoting *Nautilus*, 572 U.S. at 911). Instead, the claims must “provide objective boundaries to a POSA.” *Id.* A claim can lack “reasonable certainty” for a variety of reasons and can be found indefinite “if the patent does not convey with reasonable certainty how to measure a claimed feature,” *Teva Pharms. USA, Inc. v. Sandoz, Inc.*, 789 F.3d 1335, 1341 (Fed. Cir. 2015), “if the claim language might mean several different things and no informed and confident choice is available among the contending definitions,” *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1371 (Fed. Cir. 2014) (quotation marks omitted) (quoting *Nautilus*, 572 U.S. at 911 n.8), or if a claim term “does not have a proper antecedent basis where such basis is not otherwise present by implication,” *Halliburton Energy Servs., Inc. v. M-I LLC*, 514 F.3d 1244, 1249 (Fed. Cir. 2008). A claim can also be found indefinite if it is grammatically incoherent, *Trusted Knight Corp. v. IBM*, No. 14-1063, 2015 WL 7307134, at \*6 (D. Del. Nov. 19, 2015) (claim term was missing a verb and was not amenable to correction), *aff’d sub nom.*, 681 F. App’x 898 (Fed. Cir. 2017) or if the patent “recites function without reciting sufficient structure for performing that function,” *see Williamson v. Citrix Online, LLC*, 792 F.3d 1339 (Fed. Cir. 2015). As explained below, the VMware patents suffer from each of these infirmities, and ultimately fail to “afford clear notice of what is claimed.” *Nautilus*, 572 U.S. at 909. Indefiniteness must be proven by clear and convincing evidence. *See BASF Corp. v. Johnson Matthey Inc.*, 875 F.3d 1360, 1365 (Fed. Cir. 2017).

### III. DENSIFY’S LEVEL OF ORDINARY SKILL IN THE ART

The appropriate level for one of ordinary skill in the art as of the date of the invention of the VMware Patents-in-Suit is an individual with at least a bachelor’s degree in Computer Engineering, Computer Science or comparable degree and two years of experience developing software tools and/or computer systems for use in the area of information technology infrastructures and utilities in monitoring and managing virtualized environments. Madisetti Dec. ¶ 19.

### IV. DISPUTED CONSTRUCTIONS

#### A. U.S. Patent No. 8,336,049 (“the ’049 patent”)

##### 1. “provisioning [virtual machines]” (Claims 1, 10, and 16)

VMware’s Construction	Densify’s Construction
This term should not be construed or should be given its plain and ordinary meaning. In the alternative, this construction should be “supplying and configuring [virtual machine] computing resources.”	“instantiating, starting, stopping, suspending and de-allocating, or similar activity”

##### a. VMware’s Opening Position<sup>1</sup>

“Provisioning” is a well-known word needing no construction. In the computing context, provisioning is “[t]he act of supplying and configuring computing resources.” (*Dictionary of Computing and Internet Terms* (1st ed. 2016); see also *Merriam-Webster Dictionary*, <https://www.merriam-webster.com/dictionary/provision> (last accessed May 19, 2020) (“provision,” “provisioning”: “to supply with needed materials”; “to supply with provisions”); *Oxford Advanced Learner’s Dictionary*, [https://www.oxfordlearnersdictionaries.com/us/definition/english/provision\\_2](https://www.oxfordlearnersdictionaries.com/us/definition/english/provision_2) (last accessed May 19, 2020) (“provision”: “to supply

<sup>1</sup> VMware has slightly revised its proposed alternative construction from its construction in the Joint Claim Construction Chart (D.I. 737-1, Ex. A at 3).

somebody/something with enough of something . . .”).) If the Court adopts a construction, however, it should adopt VMware’s plain meaning and reject Cirba’s attempt to limit the term to the actions expressly recited in dependent claim 7.

VMware’s position finds support in the claims. Claims 1, 10, and 16 recite generating a prediction of future virtual machine (“VM”) utilization, receiving VM *resource utilization* data, adjusting the prediction, and provisioning VMs “in accordance to” the adjusted prediction. Based on this plain language, “provisioning” includes supplying *computing resources* sufficient to handle the predicted utilization. The dependent claims confirm this. Claims 3 and 5 state that a “metric corresponding to said provisioning” is “an amount of computing resources used.” Resources in claim 6 “include virtual machine memory and virtual processor utilization.”

The specification confirms that “provisioning” involves various ways of supplying and configuring VM resources, not just the examples in claim 7. (*See, e.g.*, ’049 patent, 1:26-28 (“enterprises using cloud computing . . . rent and provision computing resources to accommodate peak utilization requirements.”).) “Provisioning” VMs gives the VMs resources, based on “a customer’s particular [resource] needs,” required to run services. (*Id.*, 3:12–18; *see also* FIG. 5.)

VMware’s position is also consistent with the discussion of “provisioning” in the file history’s cited prior art. *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1317 (Fed. Cir. 2005). During prosecution, the Examiner raised the Wood reference (D.I. 737-3 (Ex. I) at 23–28 (VMW00099595 to -600)), which discloses “provisioning techniques to determine the resource needs of overloaded VMs.” (*Id.* at 72.) As the Wood reference explains, its “provisioning component needs to estimate the peak CPU, network and memory requirement of” such VMs. (*Id.* at 75.) It further explains that moving a VM to a server “is feasible only if that server has sufficient idle CPU, network and memory resources to meet the desired *resource allocation of*

*the candidate VM as determined by the provisioning component.*” (*Id.* at 77 (emphasis added).)

Cirba’s restrictive construction is flawed. Claims should “not be read restrictively[,]” absent “a clear intention to limit the claim scope using ‘words or expressions of manifest exclusion or restriction.’” *Liebel-Flarsheim Co. v. Medrad, Inc.*, 358 F.3d 898, 906 (Fed. Cir. 2004) (citation omitted). No such intention is present here. Instead, the specification lists “instantiate, start, stop, suspend and de-allocate” as *non-exhaustive examples* of provisioning, prefacing them with the phrase “e.g.”, and post-facing them with “etc.” (’049 patent, 3:7–13). This “leav[es] room for other types” of provisioning, as supported by the intrinsic record discussed above. *Uniloc USA, Inc. v. Microsoft Corp.*, 290 F. App’x 337, 343 (Fed. Cir. 2008).

Claim differentiation also proscribes Cirba’s attempt to limit the scope of the independent claims to the specific examples of “provisioning” recited in dependent claim 7. The presumption against imposing the claim 7 limitation is “especially strong” here, as “[it] is the only meaningful difference between [the] independent and dependent claim.” *Acumed LLC v. Stryker Corp.*, 483 F.3d 800, 806 (Fed. Cir. 2007). In this context, the open-ended “etc.” in Cirba’s construction “brings no additional clarity.” *Affinity Labs of Tex., LLC v. Clear Channel Broad., Inc.*, No. 12-205, 2014 WL 1699063, at \*10 (W.D. Tex. Apr. 29, 2014). In fact, it ignores that the examples in claim 7 already appear after the open term “comprises at least one of.”

#### **b. Densify’s Responsive Position<sup>2</sup>**

“The specification acts as a dictionary when it expressly defines terms used in the claims ....” *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996). Here, the specification defines the term “provision [virtual machines]” as “provision (*e.g., instantiate, start, stop, suspend and de-allocate, etc.*) virtual machines ....” ’049 patent at 3:8-9 (emphasis added).

---

<sup>2</sup> Densify revised its proposed construction from the Joint Claim Construction Chart (previously, “instantiating, starting, stopping, suspending, and de-allocating, etc.”). D.I. 737-1, Ex. A at 1.

“The use of ‘etc.’ in [a] definition implies additional, *but similar* forms of expression.” *Indacon, Inc. v. Facebook, Inc.*, 824 F.3d 1352, 1355 (Fed. Cir. 2016). The Court’s decision in *Arendi* is instructive. In construing “document,” which was defined in the specification as “word processors, spreadsheets, etc.,” the Court applied *Indacon* to find that “document” means “a word processing, spreadsheet, or similar file.” *Arendi*, 2019 WL 3891150 at \*4. Here, the proper construction of “provisioning” is “instantiating, starting, stopping, suspending, de-allocating, or similar activity.”

VMware’s proposed construction goes well beyond the specification’s definition. Nothing in the specification suggests that “provisioning [virtual machines]” includes “*configuring* [virtual machine] *computing resources*.” “Configuring” is an ambiguous term that is not similar to the activities of “instantiating, starting, stopping, suspending, or de-allocating.” The defined activities (instantiating, etc.) allow a VM to operate (or not) *in whatever configuration* it is in. There is no support for the proposition that “provisioning” a VM includes “configuring” the VM.<sup>3</sup>

VMware’s argument also finds no support in the claim language itself. While claims 1, 10, and 16 cite “receiving resource utilization statistics,” “adjusting the prediction,” and “transmitting instructions for provisioning virtual machines,” (’049 patent, cls. 1, 10, 16), nothing in the claims requires more than “instantiating, starting, stopping, suspending, de-allocating, etc.” virtual machines to meet demand based on the received resource utilization statistics.

VMware’s argument that its “position is also consistent with the discussion of ‘provisioning’ in the file history’s cited prior art” (VMW Op. at 5) likewise fails. VMware relies

---

<sup>3</sup> The only instance of “configure” in the specification, as it relates to a VM, pertains to using a prediction filter to predict resource demand. The specification teaches using a sample representing the “configuration of virtual machines to support a certain demand...for a certain instance in time” as part of the prediction filter calculation. ’049 patent, 4:7-25. This says only that a given configuration of VMs may be taken into account for predictive purposes. There is no discussion here, or anywhere else, about altering a VM’s configuration through “provisioning.”



on the Wood reference cited during prosecution, which discloses “provisioning techniques” that are used “to determine resource needs,” “to estimate” resource needs, and “to meet the desired resource allocation of the candidate VM as determined by the provisioning component.”<sup>4</sup> But none of these disclosures requires more than “instantiating, starting, stopping, suspending, de-allocating, etc.” virtual machines in accordance with resource utilization statistics. And VMware conveniently ignores the cited Mishra reference, which discloses an embodiment that includes a separate “provisioning manager” and “*configuration manager*,” (D.I. 737-3 (Ex. I) at 65 (Mishra at [0064])), and the ability to “provision, *configure* and deploy” an IT environment from a portal, (*id.* at 66 (Mishra at [0068])). In distinguishing Mishra, VMware acknowledged that provisioning and configuring resources are not the same. *Id.* at VMW00099616 (“Applicant understands Mishra to disclose... ‘provisioning engine ... able to provision and *configure* these resources....’”) (emphasis added). *Phillips*, 415 F.3d at 1317 (“the prosecution history can often inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the invention in the course of prosecution, making the claim scope narrower than it would otherwise be”) (citation omitted).

VMware’s claim differentiation argument is also without merit. “[C]laim differentiation is a rule of thumb that does not trump the clear import of the specification,” *Edwards Lifesciences LLC v. Cook Inc.*, 582 F.3d 1322, 1332 (Fed. Cir. 2009), and it “‘cannot broaden claims beyond their correct scope,’” *Curtiss-Wright Flow Control Corp. v. Velan, Inc.*, 438 F.3d 1374 (Fed. Cir. 2006). VMware’s proposed construction attempts to accomplish just such an impermissible broadening. Densify’s proposed construction, on the other hand, is consistent with the

---

<sup>4</sup> VMware cites passages from the Wood reference itself, but those passages were never expressly referenced or discussed in the actual file history.

presumption that dependent claims are narrower in scope than independent claims. The “at least one of” language of claim 7 requires that at least one of the listed examples be implicated in the claimed provisioning – Densify’s proposed construction allows the types of “provisioning” identified in the specification and those similar to them. “Configuring” is not within that group.

**c. VMware’s Reply Position**

Although Cirba has replaced “etc.” in its construction with “or similar activity,” its revised construction still imports an embodiment from the specification. (*Compare* D.I. 737-1 (Ex. A) at 2 *with* Cirba’s Resp. Claim Construction Br. (“Resp. Br.”) at 6.) Cirba never explains why the Court should limit the term to that embodiment, despite intrinsic support for a broader plain meaning. Nor does Cirba address dependent claims 3, 5, and 6’s recitation of broader subject matter than its construction allows. (*See* VMware’s Op. Br. (“Op. Br.”) at 4-5.)

Cirba places great weight on the specification’s list of certain “provisioning” acts. (Resp. Br. at 6-7) But the specification prefaces that discussion with “*e.g.*”—making clear that the identified acts are mere examples and not a definition. (Op. Br. at 5.) Cirba’s argument that “configuring” in VMware’s construction is not similar to “instantiating, starting, stopping, suspending, or de-allocating” (Resp. Br. at 6-7) thus is of no import.

Cirba both fails to address VMware’s case law and cites inapposite authority. In *Indacon v. Facebook*, the court limited the construction of the word “term” based on the specification’s use of “*i.e.*” to describe the relevant “terms.” No. SA-10-CA-00966-OLG, 2013 WL 12090032, at \*9 (W.D. Tex. Sept. 6, 2013). Whereas the “use of ‘*i.e.*’ signals an intent to define the word to which it refers,” *Edwards Lifesciences LLC v. Cook Inc.*, 582 F.3d 1322, 1334 (Fed. Cir. 2009), “*e.g.*” is merely exemplary. *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1373–74 (Fed. Cir. 2014) (contrasting “*i.e.*” and “*e.g.*”). *Indacon* thus does not support Cirba’s construction.

The same is true of this Court’s decision in *Arendi S.A.R.L. v. LG Electronics, Inc.*, No.

12-1595-LPS, 2019 WL 3891150 (D. Del. Aug. 19, 2019). There, the Court limited “document” to “word processing documents” based on a specification that “defined” the invention in that manner. *Id.* at \*3. The specification here uses no similar definitional signal.

The prosecution history also supports VMware’s construction. Contrary to Cirba’s gloss on the cited Wood reference, its “provisioning” was broader than Cirba’s construction. (*See* Resp. Br. at 7.) Wood’s “provisioning component” “estimated” and “determined” the “resource needs” of the VMs, especially their “CPU, network and memory requirement[s].” (Op. Br. at 5.)

The cited Mishra reference similarly does not support Cirba’s narrow construction. (Resp. Br. at 7-8.) Nowhere does Mishra limit “provisioning” to the specific acts in Cirba’s construction. To the contrary, consistent with VMware’s construction, the Examiner explained that “the provisioning [in Mishra] was in at least one instance for the changing of the allocation of resources.” (*See* D.I. 737-3 (Ex. I) at 25 (VMW00099597); *see also id.* at 63, 66–67 (Mishra, ¶¶ [0046], [0072]–[0075]).) Nor did VMware “acknowledge[] that provisioning and configuring are not the same,” as Cirba asserts. (Resp. Br. at 7.) Instead, VMware disputed that Mishra was relevant at all: “Mishra does not teach, describe or suggest ‘transmitting instructions relating to provisioning virtual machines for the application.’” (D.I. 737-3 (Ex. I) at 43 (VMW00099615) (emphasis in original).) VMware’s statements certainly did not “clear[ly] and unmistakabl[y]” disclaim scope. *Omega Eng’g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1325-26 (Fed. Cir. 2003).

Finally, Cirba completely ignores VMware’s dictionary definitions (Op. Br. at 4-5) that “clearly define the term.” *H. Lundbeck A/S v. Apotex Inc.*, No. CV 18-88-LPS, 2019 WL 3206016, at \*6 (D. Del. July 16, 2019) (relying on dictionary definitions).

#### **d.      Densify’s Sur-Reply Position**

“Provision,” as that term is used throughout the patent, is defined generally and is not limited to any specific embodiment. The way it is defined (“to provision (e.g., instantiate, start,

stop, suspend and de-allocate, etc.)”) mirrors the way the term “document” was defined in *Arendi S.A.R.L. v. LG Electronics, Inc.*, 2019 WL 3891150 (D. Del. Aug. 8, 2019) (J. Stark). There, the specification stated, “the present invention is applicable to all types of word processing documents *such as* NOTEPAD™, WORDPAD™, WORDPERFECT™, QUATRO-PRO™, AMIPRO™, *etc.*” *Id.* at \*3-4 (emphases added). In view of this disclosure, the Court construed “document” to mean “a word processing, spreadsheet, *or similar file* into which text can be entered.” *Id.* (emphasis added). The “such as” language of *Arendi* is the linguistic equivalent of the “e.g.” in the ’049 patent’s definition of “to provision,” and the “etc.” is identical. *See U.S. Auto. Ass’n v. Wells Fargo Bank, N.A.*, 2019 WL 2488135, at \*12 (E.D. Tex. June 13, 2019) (“[t]his is both *exemplary (‘e.g.’)* and *non-exhaustive (‘etc.’)*”; adopting Defendant’s “*such as*” language in construction) (emphases added). By this reasoning, Densify’s construction is proper.

VMware contends that the Wood reference, which was cited during prosecution, uses “provisioning” more broadly than Densify’s proposed construction. But Wood’s disclosure of “provisioning techniques” to “determine” and “estimate” resource needs is entirely consistent with the patent’s definition of “provisioning.” The same is true of the Mishra reference, which treats “provisioning” and “configuring” as distinct activities. Densify’s Resp. at 7-8.

VMware’s extrinsic evidence is inapposite. “While dictionaries may be used to ascertain the plain and ordinary meaning of claim terms, the intrinsic record is used to resolve ambiguity in claim language or, where it is clear, trump inconsistent dictionary definitions.” *W.E. Hall Co., Inc. v. Atlanta Corrugating, LLC*, 370 F.3d 1343, 1350 (Fed. Cir. 2004) (citations omitted).

## 2. “virtual machine utilization for the application” (Claims 1, 10, and 16)

VMware’s Construction	Densify’s Construction
This term should not be construed or should be given its plain and ordinary meaning. In the alternative, the construction should be	“number of virtual machines required for the application”

“utilization of virtual machine computing resources for the application.”	
---	--

**a. VMware’s Opening Position**

As with the “provisioning” term, Cirba attempts to read a limitation from a dependent claim into the independent claims. This simple term takes its plain meaning and requires no construction. In the alternative, the intrinsic record supports only VMware’s construction.

The claims recite predicting VM utilization, receiving *resource utilization* statistics, and adjusting the prediction using the received *resource utilization* statistics. This makes clear that “virtual machine utilization” refers to using VM computing resources generally. The dependent claims confirm this. Claim 3 recites applying a metric in a bill for “*use of said [VMs].*” (Emphasis added.) Per claim 5, the metric can be the “amount of computing resources used,” including memory and processor utilization as recited in claim 6.

This plain meaning is also apparent from the specification. The patent discloses that “enterprises . . . rent and *provision computing resources to accommodate peak utilization requirements.*” (’049 patent, 1:25–28 (emphasis added).) Figure 3 “depict[s] computing resource utilization,” shown as a percentage. (*Id.*, 2:52–53, FIG. 3.) And in discussing Figure 5, the specification explains that “analytics engine 405 *compares the received utilization metrics with the generated prediction* from step 510 and, in step 535, adjusts its generated prediction based upon the comparison.” (*Id.*, 3:60–63 (emphasis added).) The specification also discloses “a billing platform . . . to charge customers on a per-virtual machine/unit time basis *or resources (e.g., memory, CPU utilization, etc.) used.*” (*Id.*, 4:41–44 (emphasis added).)

Cirba attempts to limit this term to a narrow example that appears in a dependent claim—“number of virtual machines,” ignoring claim differentiation. *See Phillips*, 415 F.3d at 1315. “Number of virtual machines,” recited in dependent claim 4 as an exemplary “metric,” cannot be

read into the broader term “virtual machine utilization” of the independent claims. Cirba also ignores the broader disclosure in the specification, including the specification’s discussion of memory and CPU utilization generally and of resource utilization as a *percentage* in Figure 3.

**b. Densify’s Responsive Position**

The specification describes an analytics engine that is used “to generate a prediction of demand for the LOB [(or “line of business”)] application in order to dynamically control *the provisioning of virtual machines* to support the LOB application.” ’049 patent, Abstract (emphasis added). The specification further notes that “[s]uch an analytics engine enables an enterprise to match its computing costs on a ‘per virtual machine’ basis for the LOB application....” ’049 patent, 1:48-50 (“Summary Of The Invention”). When discussing an architecture for deploying line of business applications, the specification notes that “any *number* and tiers *of virtual machines* may be used to support the architecture....” *Id.* at 3:25-26.

VMware cites claim language and examples from the specification directed to resource utilization statistics in support of its alternative construction, but each example is merely directed to the provisioning of the *number* of virtual machines required for the application in accordance with an adjusted prediction based on resource utilization statistics, or the context of billing subsequent to the provisioning of the number of virtual machines. VMware’s citations to dependent claims fare no better, as each is directed to the separate objective of “recording a metric” corresponding to the claimed provisioning of virtual machines and applying the metric for billing purposes; these dependent claims do not speak to the step of actually provisioning virtual machines or how said provisioning is accomplished.

Claim differentiation does not aid VMware’s argument. Dependent claim 4 substantially differs from independent claim 1, at least because it adds the limitations of “recording a metric corresponding to said provisioning of virtual machines” and “applying the metric in a bill to a

customer for use of said provisioned virtual machines.” *Id.*, cls. 3, 4. Indeed, this is not a case where “the limitation in dispute is the only meaningful difference between an independent and dependent claim....” *Acumed LLC v. Stryker Corp.*, 483 F.3d 800, 806 (Fed. Cir. 2007).

**c. VMware’s Reply Position**

Cirba does not explain why the Court should narrow the plain meaning of this independent claim limitation to an embodiment in the specification or to a “metric” limitation from dependent claim 4. (Resp. Br. at 13.) To argue otherwise, Cirba ignores portions of the intrinsic record that do not fit its narrow construction.

Cirba baldly asserts that “each example [that VMware cites] is merely directed to the provisioning of the *number* of virtual machines.” (*Id.* at 13.) But as VMware has explained, the plain meaning and intrinsic evidence refute this. (Op. Br. at 11–12.) For example, dependent claims 5 and 6 recite “virtual machine utilization” reflecting the “amount of computing resources,” “including memory and processor utilization.” These examples go beyond the mere “number of virtual machines.” Cirba also would exclude (i) VM utilization as a *percentage* of resources and (ii) “memory, CPU utilization, etc.” (’049 patent, FIG. 3; 2:52–53, 4:41–44.)

Cirba’s argument also confuses the claim step at issue. Although Cirba argues that “the[] dependent claims do not speak to the step of actually provisioning virtual machines” (Resp. Br. at 13), this disputed term appears in claim 1’s “generating” step, not its “provisioning” step. Cirba’s overly narrow construction of “provisioning” provides no support for its overly narrow construction of “virtual machine utilization.” (*Id.*)

**d. Densify’s Sur-Reply Position**

Densify’s construction is supported by the specification (’049 patent, 1:48-50; 3:25-26; 4:39-49) and may be consistently interpreted across all claims. Assuming Densify’s construction, the first element of claim 1 recites “generating a prediction of future *number of virtual machines*

*required....*” Claim 3, which depends from claim 1, requires “*recording* a metric corresponding to” the “provisioning virtual machines” recited in the last element of claim 1. Claim 3 thus adds a “recording” requirement, which is an additional activity not claimed in claim 1. *What* gets recorded is not specified; the only requirement is that it “correspond[]” to “provisioning virtual machines.” Densify’s construction therefore creates no inconsistency with respect to claim 3. Claim 4 depends from claim 3 and requires that the “metric” be “a number of virtual machines instantiated by said customer.” The added limitation requires that the recorded metric be the “number of virtual machines instantiated by the customer” – there is no requirement that this be the “number of virtual machines required for the application” in the “generating” step of claim 1. The claims themselves prove false VMware’s claim differentiation argument.

**3. “receiving/received resource utilization statistics” (Claims 1, 8–10, 13, 14, 16, 19, and 20)**

VMware’s Construction	Densify’s Construction
This term should not be construed or should be given its plain and ordinary meaning. In the alternative, the construction should be “receiving/received past or current resource utilization statistics.”	“receiving real-time resource utilization statistics” / “received real-time resource utilization statistics”

**a. VMware’s Opening Position<sup>5</sup>**

Here, too, Cirba improperly adds a limitation, “real-time,” from the specification into the claims. But the simple phrase “receiving resource utilization statistics” needs no construction. In the alternative, VMware’s construction is the only one supported by the intrinsic record. The steps of claims 1, 10, and 16 involve generating a prediction, “receiving resource utilization statistics,” and then adjusting the prediction using those received statistics. No claim language requires adjusting the prediction using only “real-time” statistics. In fact, the patent discloses a

---

<sup>5</sup> VMware has slightly revised its proposed alternative construction from its construction in the Joint Claim Construction Chart (D.I. 737-1, Ex. A at 3) to bring the parties’ positions closer.



broader embodiment, in which “the resource utilization metrics received” are used for a prediction filter and include “past, present, and future samples.” (’049 patent, 4:7–20; *see also id.*, claims 9, 14, 20.) Receiving “past” samples before adjusting a prediction using the filter is consistent with VMware’s construction and refutes Cirba’s improper narrowing.

### b. **Densify’s Responsive Position**

The issue here is whether the “resource utilization statistics” in the “receiving” step must be limited to “real-time” (or current) statistics, or if they may also include “past” statistics. To begin, the parties agree that the steps of claim 1, 10, and 16 are “ordered/sequenced,” meaning they must occur in the order listed: the “generating” step must precede the “receiving” step, etc. D.I. 737-1, Ex. A at 1. The claim language makes plain that “past resource utilization statistics” are necessary to the “generating” step (“generating a prediction of future virtual machine utilization for the application *based upon past resource utilization statistics*”). This must mean that “past resource utilization statistics” were available in the “generating” step which took place *before* the “receiving.” Common sense dictates that there is no import to receiving what was already available, making receiving “past resource utilization statistics” *after* having already used them in the “generating” step illogical. Further, VMware’s construction allows the “receiving” step to receive *only* “past resource utilization statistics” (“receiving past *or* current resource utilization statistics”), which would render the claim invalid for lack of written description; nowhere in the specification is it taught that the “prediction” generated in the “generating” step can be adjusted in the “adjusting” step using nothing but *past* resource utilization statistics.<sup>6</sup>

---

<sup>6</sup> VMware appears to consider the term “current” to be synonymous with “real-time,” *i.e.*, reflective of what is occurring with respect to VM utilization at the current point in time. The specification, however, chooses the term “real-time” to connote present utilization.<sup>6</sup> ’049 patent, Abstract (“An analytics engine receives *real-time* statistics”); Fig. 5 (515); 3:2-7 (“periodically

**c. VMware’s Reply Position**

Cirba’s construction would limit this term to “real-time” statistics, importing an example from the specification into the claims. But Cirba does not address the patent’s prediction filter, in which “the resource utilization metrics received” include “past, present, and future samples.” (Op. Br. at 15 (citing ’049 patent, 4:7-20, cls. 9, 14, 20).) VMware, by contrast, makes clear that the claim language does not limit the received statistics to any specific time—past or current.

Cirba is wrong that VMware’s construction is somehow “illogical.” (Resp. Br. at 16.) Although some “past resource utilization statistics” are received in the “generating” step, there may be good reason to receive additional past utilization data in the receiving step. For example, the method may “generate” a prediction based on day-old resource utilization statistics, followed by “receiving” more current (but not necessarily “real time”) statistics to adjust the prediction. The disclosed embodiment with its prediction filter describes this use. (Op. Br. at 15.)

Cirba’s written description argument (Resp. Br. at 16) has no bearing on claim construction and lacks merit. VMware’s construction is fully supported by the prediction filter in the specification, which adjusts predictions based on utilization statistics that are not limited to just “real time” data. (*See* ’049 patent, 4:7-20.)

**d. Densify’s Sur-Reply Position**

VMware relies on the specification’s prediction filter to oppose Densify’s construction (VMW Reply at 16-17), but fails to address (1) that the prediction filter applies only to “one embodiment,” (’049 patent, 4:7-22) and therefore is not properly imported into a claim; and (2) the prediction filter calls for “*future* samples of an interval of time” (as well as “past” and “present” samples) but “future” statistics is nowhere in VMware’s proposed construction. An embodiment

---

transmit *real-time* performance statistics to analytics engine 405”); 4:2-6 (“adjusting predicted demand in accordance to *real-time* actual utilization feedback from the virtual machines”).

disclosed in the specification is not justification for its (partial) importation into construction of the “receiving” step. Rather, Fig. 5, which is a flow chart essentially depicting claim 1 (*id.* at 2:7-9), the specification, and the claim language itself, all dictate that the “statistics” (or “metrics”) received in the “receiving” step are “real-time” statistics. *Densify Resp.* at 15-16.

**4. “transmit[ting] instructions relating to provisioning virtual machines”  
(Claims 1, 10, and 16)**

<b>VMware’s Construction</b>	<b>Densify’s Construction</b>
Not indefinite. This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

**a. VMware’s Opening Position**

Cirba’s indefiniteness position lacks merit. Once the dispute over “provisioning” is resolved, the simple English phrase “transmitting instructions related to provisioning” provides reasonable certainty as to claim scope. The claims and specification describe “transmitting instructions” to provision VMs so they can tackle “future demand . . . in accordance with the adjusted generated prediction.” (’049 patent, 3:64–67, claims 1, 10, 16; *see also id.*, FIGS. 4, 5; 3:7–12 (“Analytics engine 405 communicates with service manager 420 to instruct service provider 400 to provision . . . virtual machines in accordance with predicted demand . . . as derived by analytics engine 405.”).) There is no difficulty understanding such “instructions.”

The specification also explains that instructions could ensure—for example through a service library—adequate VM resources to service the predicted demand:

Upon receiving such instructions from analytics engine 405, service manager 420, in turn, communicates with service library 452, which stores the relevant technology (e.g., platform, software, data, etc.) that is needed to provision a particular virtual machine (e.g., to run an application server, database server, directory server, web server, etc.) for a customer’s particular needs.

(*Id.*, 3:12–18; *see id.*, 3:64–4:2.) This use of the instructions confirms the term’s clear meaning.

**b. Densify’s Responsive Position**

Claims 1, 10, and 16 each require “transmit[ting] *instructions* relating to provisioning virtual machines,” but the term “instructions” has no antecedent basis. The claim neither explicitly nor implicitly indicates *what* the instructions are or *how* the instructions are determined.

The term “instructions” has no predicate in the claim. It appears for the first and only time in the “transmitting” step, which recites that the instructions “relat[e] to provisioning virtual machines . . . in accordance with the adjusted prediction.” ’049 patent, cl. 1. But nowhere in the claim is it recited how the instructions are generated or how they relate to the adjusted prediction. The specification provides little guidance. Two references in the specification discuss an analytics engine “instructing” a service provider to “provision virtual machines,” but provide insufficient information on what those instructions are or how they are generated.<sup>7</sup> ’049 patent, 3:7-11 (“Analytics engine 405 communicates with service manager 420 to *instruct* service provider 400 *to provision* (e.g., instantiate, start, stop, suspend and de-allocate, etc.) virtual machines in accordance with predicted demand for the LOB application as derived by analytics engine 405); 3:64-67 (“*instructions to provision* virtual machines”) (emphases added).

**c. VMware’s Reply Position**

Cirba concedes that “instructions *to provision*” would not be indefinite. (Resp. Br. at 18-19 n.7.) There is no reason for the Court to find “instructions relating to provisioning” any more indefinite. Contrary to Cirba, a skilled artisan would understand, based on the intrinsic record, “what those instructions are [and] how they are generated.” (*Id.* at 18-19.) These instructions ensure the VMs have adequate resources to handle the adjusted predicted utilization. (Op. Br. at

---

<sup>7</sup> Densify considered the construction “more than one instruction to instantiate, start, stop, suspend, de-allocate” or the like, but the claim language appears to foreclose that possibility. Had the claim read, “instructions *to provision*,” this may have been a proper construction. But because the claim recites “instruction *relating to* provisioning,” it is unclear how they are related.

18; '049 patent, cl. 1, 3:7-12, FIG. 5 (exemplary “process of transmitting instructions to service manager 420 in accordance with a predicted demand,”) 3:30-32, 3:64-67.) The specification discloses an example in which the method transmits instructions relating to provisioning VMs to a service library, to ensure adequate VM resources to service predicted demand. (*Id.* at 3:12–18.)

Without legal authority or evidentiary support, Cirba asserts that “instructions” requires an “antecedent basis” to avoid indefiniteness. (Resp. Br. at 18-19.) Cirba is incorrect; as the term “instructions” appears for the first time in the transmitting step, it needs no antecedent basis. *See Microprocessor Enhancement Corp. v. Tex. Instruments Inc.*, 520 F.3d 1367, 1379 (Fed. Cir. 2008) (claim term appearing for first time without definite article or other qualifier “neither had nor required an antecedent basis”). But even if otherwise, “[a] failure to provide [one] does not necessarily render a claim indefinite” as long as the skilled artisan can “discern the boundaries on the claim based on the [intrinsic record] and the knowledge in the relevant art.” *Yodlee, Inc. v. Plaid Techs., Inc.*, No. 14-1445-LPS, 2016 WL 204372, at \*12 (D. Del. Jan. 15, 2016) (internal quotations and citations omitted). As discussed above, the boundaries here are clear.

“[T]he prosecution history [also] shows the patent examiner had no trouble analyzing the term in relation to [the] prior-art.” *3G Licensing, S.A. v. Blackberry Ltd.*, No. 17-82-LPS, 2018 WL 4375091, at \*8 (D. Del. Sept. 13, 2018); *see Sonix Tech. Co. v. Publ’ns Int’l, Ltd.*, 844 F.3d 1370, 1380 (Fed. Cir. 2017) (no Patent Office uncertainty as to scope of “visually negligible” and no apparent difficulty applying limitation to prior art). Here, the Examiner had no difficulty understanding “provid[ing] provisioning instructions in accordance with an adjusted prediction” when comparing the claim to the prior art. (D.I. 737-3 (Ex. I) at 25 (VMW00099597).)

**d.      Densify’s Sur-Reply Position**

The only thing a POSA knows about the “instructions” transmitted via the “transmitting” step is that they must “relat[e] to provisioning virtual machines ... in accordance with the adjusted prediction.” Nothing is recited regarding what the instructions are, how they are determined, or in what way(s) they “relate” to the adjusted prediction. Must the claimed “instructions” be instructions to provision (“instantiate, start, stop, suspend or de-allocate”) virtual machines, as the specification discloses? ’049 patent, 3:7-18; Disputed Term A. Are instructions simply “relating” to such provisioning activities within the scope? And how attenuated may the relationship to provisioning be? This element lacks reasonable certainty and is therefore indefinite. *See Elm 3DS Innovations, LLC v. Samsung Elecs. Co.*, 2020 WL 1850657, at \*3 (D. Del. Apr. 13, 2020).

VMware’s citations to *Yodlee, Inc. v. Plaid Techs., Inc.*, No. 14-1445-LPS, 2016 WL 204372, at \*12 (D. Del. Jan. 15, 2016) and *Microprocessor Enhancement Corp. v. Tex. Instruments Inc.*, 520 F.3d 1367, 1379 (Fed. Cir. 2008) are inapposite. *Microprocessor* concerns an amendment made during prosecution that resolved an indefiniteness issue by the Examiner; VMware’s citation to a parenthetical within that analysis is untethered from any issue here. *Yodlee* reinforces *Densify*’s position by finding that indefiniteness is not appropriate where a POSA *can* “discern the boundaries of the claim” – here, they cannot. And VMware’s arguments that the “transmit[ing]” term cannot be indefinite because “a patent examiner reviewed the term and did not object on the basis of indefiniteness” are “ultimately unconvincing.” *Avenue Innovations, Inc. v. E. Mishan & Sons Inc.*, 310 F. Supp. 3d 457, 465 (S.D.N.Y. 2018).

**B. U.S. Patent No. 8,875,266 (“the ’266 Patent”)****1. “identifying” (Claim 1)**

<b>VMware’s Proposed Construction</b>	<b>Densify’s Proposed Construction</b>
This term should not be construed or should be given its plain and ordinary meaning.	“specifically or particularly indicating”

**a. VMware’s Opening Position**

“Identifying” is a simple English word that needs no construction.

Cirba’s proposed construction grafts a “specifically or particularly” limitation into the claim that is contrary to the ordinary meaning of the word “identifying”: “establish or indicate who or what (someone or something) is.” (“Identify,” *New Oxford Am. Dictionary* 839 (2d ed. 2005).) Cirba’s construction also lacks support in the specification. The specification does not define “identifying,” let alone proscribe that it be “specific[] or particular[],” because it does not need to do so. Claim 1’s plain language is already clear by requiring that the “policies identify[] one or more hardware platforms.” (*See also, e.g.,* ’266 patent, 9:1-14, 9:40-57, 10:22-37, 13:14-54 (exemplifying “identifying” in the context of the claims).) *See IOEngine, LLC v. Interactive Media Corp.*, No. 14-1571-GMS, 2016 WL 1121938, at \*1, n.4 (D. Del. Mar. 21, 2016) (“[T]he court declines to construe terms that are already evident from reading the entire claim.”) (citing *Kyocera Wireless Corp v. ITC*, 545 F.3d 1340, 1347-49 (Fed. Cir. 2008)).

**b. Densify’s Responsive Position**

Claim 1 deals only with whether a virtual machine is permitted to execute on the host platform on which it currently resides. Although the preamble recites “a method of securely controlling execution of *a computer program* within a virtual machine,” the elements of the claim involve *only* whether or not the *virtual machine* can execute on a particular host, not whether the

“computer program” is (or is not) authorized to execute on that virtual machine.<sup>8</sup> ’266 patent, cl. 1 (“whether the *current host platform matches one of the hardware platforms* identified by the policies” “*for which the virtual machine is authorized to execute*”). The specification teaches that in order to assess whether there is a “match” between the current host platform and the one or more identified hardware platforms, it is necessary to identify the one or more hardware platforms with specificity. ’266 patent, 1:19-25 (noting interest in “restricting of software to *particularly identified* system hardware or computer platforms”); 1:27-31 (noting need for system managers “in controlled, secure environments,...to...ensure that...particular software is executable on *particular systems*”); 9:62-65 and 13:35-41 (discussing property files 170, which are stored in “policy control files 100,” and “license rights to use a specific set of application features on a *specific host platform*”); 14:53-58 (“*unique identification* of the allowed host platforms”).

Even an “ordinary meaning” construction requires particularity. VMW Op. at 21 (the ordinary meaning of “identify” is to “establish or indicate who or *what* (someone or *something*) *is*”). VMware’s “plain and ordinary” construction differs only in that it uses vernacular other than what is in the specification.

### c. VMware’s Reply Position

Cirba offers no reason to construe the common English word “identifying.” Cirba points to no evidence of confusion about its meaning, nor does Cirba cite any reason in the intrinsic record for qualifying “identifying” by requiring a “specific” or “particular” degree of indication.

Although Cirba cites particular uses of the term “identify” in the specification, they refute its position. (Resp. Br. at 22-23.) The specification’s use of the phrases “*particularly identified systems*” and “*unique identification*,” (*id.*), confirm that the term “identifying” by itself does not

---

<sup>8</sup> Hence claim 1 does not require “software license compliance” with respect to the computer program – that is not claimed until Claim 3, which is not asserted.



connote these degrees of particularity. If otherwise, the specification would not need to add such modifiers to forms of the word “identify.” *Jazz Pharm., Inc. v. Amneal Pharm., LLC*, 895 F.3d 1347, 1361-62 (Fed. Cir. 2018) (“The claim language simply refers to ‘prescription requests containing *information identifying* patients . . . .’ It recites no specific types of identifying information.”) (emphasis original).

#### **d.      Densify’s Sur-Reply Position**

The issue is whether the “one or more hardware platforms” in claim 1 must be identified with particularity. VMware resists expressly requiring such specificity, despite repeated supporting disclosures in the specification. ’266 patent, 1:19-25 (“restricting of software to *particularly identified* system hardware or computer platforms”); 1:27-31 (noting need “to...ensure...particular software is executable on *particular systems*”); 9:62-65 and 13:35-41 (discussing “license rights to use a specific set of application features on a *specific host platform*”); 14:53-58 (“*unique identification* of the allowed host platforms”). VMware’s plain and ordinary definition requires particularity (VMW Op. at 21 (“establish or indicate who or *what* (someone or *something*) is”), but VMware also argues that “‘identifying’ by itself does not connote these degrees of particularity” (VMW Reply at 23), thereby signaling an intent to read the term more broadly.

VMware’s citation to *Jazz Pharms., Inc. v. Anneal Pharms., LLC*, 895 F.3d 1347 (Fed. Cir. 2018) is readily distinguishable. The issue in *Jazz* is whether the term “information” should be construed to mean at least particular types of information. *Id.* at 1361. It has nothing to do with construction of the term “identifying.”

#### **2.      “prohibiting the virtual machine from executing” (Claim 1)**

<b>VMware’s Proposed Construction</b>	<b>Densify’s Proposed Construction</b>
This term should not be construed or should	“terminating the virtual machine, or

be given its plain and ordinary meaning.	preventing the virtual machine from initializing or completing initialization”
--	--

**a. VMware’s Opening Position**

This phrase, including the term “prohibiting,” is plain English and requires no further construction, given that “the virtual machine” is software that may be executed.

Cirba’s construction arbitrarily limits the plain meaning of “prohibiting . . . from executing” to “terminating . . . , or preventing . . . from initializing.” But nothing in the specification defines “prohibiting” this way or limits the manner in which the claimed method “prohibit[s] the virtual machine from executing” as in Cirba’s proposal. Moreover, the patentee has not shown a clear intention to limit claim scope using “words or expressions of manifest exclusion or restriction.” *Liebel-Flarsheim*, 358 F.3d at 906 (citation omitted).

**b. Densify’s Responsive Position<sup>9</sup>**

The specification does not use the term “prohibit” to describe what happens to a virtual machine that is not authorized to operate on a host platform – rather, it repeatedly uses the term “terminate.” ’266 patent, Fig. 4 (*TERMINATE VM* (162) where execution is not permitted; VM initialization not permitted to continue (154 and 168)); 9:54-57 (“If authentication fails, virtual policy control layer 102 returns an indication of authentication failure...which then *terminates* further execution of virtual machine 64”). The plain and ordinary definition of “prohibit” does not adequately convey what the patentees have effectively defined the term to mean.

**c. VMware’s Reply Position<sup>10</sup>**

“Prohibit” is ordinary English and requires no construction. Cirba contends that “the

<sup>9</sup> Densify has slightly revised its proposed construction from its construction in the Joint Claim Construction Chart from “terminating the virtual machine, or preventing the virtual machine from initializing” (D.I. 737-1, Ex. B at 6) to “terminating the virtual machine, or preventing the virtual machine from initializing or completing initialization.”

<sup>10</sup> Cirba altered its proposed construction to append “preventing the virtual machine from . . .

patentees have effectively defined the term,” but points to nothing in the patent that constitutes a definition. (Resp. Br. at 25.) “Absent [such] disclaimer or lexicography, the plain meaning of the claim controls.” *Toshiba Corp. v. Imation Corp.*, 681 F.3d 1358, 1369 (Fed. Cir. 2012).

Cirba also incorrectly asserts that the specification “does not use the term ‘prohibit’ to describe what happens to a virtual machine that is not authorized to operate.” (Resp. Br. at 25.) The specification discloses that the “security policy controller may . . . *prohibit* the virtual machine from executing on the current host platform when the policies do indicate that the virtual machine is permitted to execute.” (’266 patent, 4:62-67 (emphasis added).)

Cirba relies heavily on embodiments that use the term “terminate.” (Resp. Br. at 25.) But these are examples, not definitions; nothing limits the claim to those examples. *See Hill-Rom Servs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1371-72 (Fed. Cir. 2014) (disclosure of only one embodiment still insufficient to limit claim scope absent manifest exclusion or restriction).

#### **d.      Densify’s Sur-Reply Position**

VMware is correct that the specification uses the term “prohibit” to describe what happens to an unauthorized virtual machine – it does so *once*. ’266 patent, 4:62-67 (“security policy controller may . . . prohibit the virtual machine from executing....”). The other *eleven* times the specification discusses what happens in the event of unauthorized or invalid operation, it uses the term “terminate.” *Id.*, Figs. 4, 6, 7, 10; 9:54-58; 10:43-49; 10:66-11:4; 11:8-11; 12:19-21; 14:30-32; 14:38-40. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1315 (Fed. Cir. 2005) (“[C]laims must be read in view of the specification, of which they are part.”). Plain and ordinary meaning is too vague: “prohibit” is defined as “to forbid by authority” or “to prevent from doing something,” but neither is pertinent in view of the specification. *See* <https://www.merriam-webster.com/dictionary/>

---

completing initialization.” (*Compare* D.I. 737-1 (Ex. B) at 6 *with* Resp. Br. at 25.) It cites no support in the intrinsic record for its proposed limitation.

prohibit (last accessed July 7, 2020). *Hill-Rom Services, Inc. v. Stryker Corp.*, 755 F.3d 1367, 1373 (Fed. Cir. 2014), where the court rejected a construction based on a preferred embodiment, is inapposite. Here, “prohibit” and “terminate” are used to describe the invention generally.

**3. “permitting the virtual machine to execute” / “authorized to execute” (Claim 1)**

<b>VMware’s Proposed Construction</b>	<b>Densify’s Proposed Construction</b>
This term should not be construed or should be given its plain and ordinary meaning.	“permitting the virtual machine to initialize or continue running” / “permitted to initialize or run”

**a. VMware’s Opening Position**

The parties’ disputes about these limitations are the same as their dispute with respect to the “prohibiting the virtual machine from executing” limitation. (*See supra* § IV.B.2.a.) For the same reasons, the claim phrases are clear on their face and need no further construction.

**b. Densify’s Responsive Position**

These phrases should be treated synonymously and should be construed consistently with Disputed Term B.2, for the reasons stated therein.

**c. VMware’s Reply Position**

The parties agree that the Court should construe these phrases consistently with “prohibiting the virtual machine from executing.” (Resp. Br. at 27.) None requires construction.

**d. Densify’s Sur-Reply Position**

The parties agree these terms should be construed consistently with Disputed Term B.2.

**4. “policies” (Claim 1)**

<b>VMware’s Proposed Construction</b>	<b>Densify’s Proposed Construction</b>
“data defining access and/or operating controls on the operation of virtual machines on host platforms”	“data identifying one or more hardware platforms for which the virtual machine is authorized to execute”

**a. VMware’s Opening Position**

The specification describes policies, stating “[p]olicy control files 100 are used to store information defining access and operation controls on the operation of the virtual machine” operating on host platforms. (’266 patent, 6:41-49.) For instance, policies “may reference or define the applicable access controls that will allow access from virtual machines to license files” (*id.*, 7:24-29) or “identify configuration parameters [*i.e.*, operating controls] on” VMs operating on host platforms (*id.*, 9:58-61). VMware’s construction reflects these teachings.

This term is also part of a larger limitation in claim 1: “the policy enforcer accessing policies . . . , the policies identifying one or more hardware platforms for which the virtual machine is authorized to execute.” While claim 1 relies on policies for a particular task—*i.e.*, “identifying one or more hardware platforms for which the virtual machine is authorized to execute,” per above, the specification confirms that “policies” are not so narrowly defined.

Cirba’s proposal is improper for reasons beyond its narrowness. Cirba does not define “policies” at all. Instead, it merely replaces “data” for “policies” in claim 1 and adds other words already appearing in claim 1. Cirba’s proposal renders those other words superfluous. This cannot be correct, as a proper “claim constructions give[s] meaning to all of a claim’s terms” and avoids redundancy. *Apple, Inc. v. Ameranth, Inc.*, 842 F.3d 1229, 1236-37 (Fed. Cir. 2016).

**b. Densify’s Responsive Position**

The purpose of the ’266 patent is to “enforce” software license compliance by using “policies” to identify “one or more hardware platforms for which the virtual machine is authorized to execute.” *See, e.g.*, ’266 patent, Title (“System and Methods for **Enforcing Software License Compliance** With Virtual Machines”) (emphasis added); Summary (noting the “policies identify[] one or more hardware platforms for which the virtual machine is authorized to execute”) (emphasis added). Where the policies authorize execution, the virtual machine is permitted to run on the host

platform; where it is not, the virtual machine is terminated or not permitted to initialize (*see* Disputed Term B.2, *supra*). The specification and claim language define “policies” as “identifying one or more hardware platforms for which the virtual machine is authorized to execute.” ’266 patent, Summary (4:60-62) and cl. 1 (“the policies identifying one or more hardware platforms for which the virtual machine is authorized to execute”).

The parties agree that “policies” are “data,” but differ as to how to define what that data does. Densify’s proposed construction is a verbatim recitation of the specification’s definition of the term. Although VMware criticizes Densify for using “words already appearing in claim 1” in its proposed construction, this Court has determined that doing so for clarity is acceptable. *Arendi*, 2019 WL 3891150 at \*6-7. VMware’s construction is not apt because it defines “policy control files,” not the “policies” themselves. ’266 patent, 6:47-49 (“Policy control files 100 are used to store information defining access and operating controls on the operation of the virtual machine.”). Moreover (and critically), VMware attempts to broaden the claim by adding an “or” between “access” and “operating controls.” There is no support in the specification for this. As noted, “policy control files” is defined as “stor[ing] information defining access *and* operating controls” (’266 patent, 6:47-49) (emphasis added)), not “access *and/or* operating controls.” Nowhere is there support for the possibility that policies can consists solely of “operating controls.”

### c. VMware’s Reply Position

Although Cirba asserts that “VMware’s construction is not apt,” Cirba acknowledges that VMware bases its construction on the specification’s discussion of “policy control files,” which contain policies. (Resp. Br. at 28-29.) The use of “and/or” in VMware’s construction is appropriate because the patent’s examples make plain that policies need not define *both* access and operating controls. The patent discloses “software 115 *may* implement a virtual policy enforcer layer 102 that controls access to encrypted virtual machine folder 96 dependent on *the*

*access control policies maintained within the policy control files 100.*” (’266 patent, 6:50-62 (emphasis added).) It also teaches that “[i]n one embodiment, policy control files 100 store one or more properties files 170 . . . that define the *operating parameters of the virtual machine 64 and application 62.*” (*Id.*, 9:58-64 (emphasis added).)

Cirba’s proposed construction, by contrast, does not actually define “policies.” Instead, it takes the word “data” and then parrots other language already appearing in the claim. While Cirba contends that its “proposed construction is a verbatim recitation of the specification’s definition of the term” (Resp. Br. at 28), it is not a definition. Cirba’s proposal simply reflects one claimed characteristic of the policies without defining what a “policy” is. (*Id.*)

#### **d.      Densify’s Sur-Reply Position**

VMware’s proposed construction equates “policies” with “policy control files,” even though the specification makes clear (and VMware does not dispute) that they are not the same: “policies” are contained within, and are only *part* of what comprises, “policy control files.” ’266 patent, 6:50-57 (“access control policies maintained within the policy control files 100”); 9:9-12 (“A copy control list 118 (FIG. 2) is preferably embedded within the policy control files 100 to store identifiers used to control derivative distribution of application 62.”). Moreover, policies must relate to controlling “access,” which is the heart of the invention (*Id.*, Title, Abstract, Claim 1), and cannot consist solely of “operating controls,” which VMware’s unsupported and overbroad “and/or” construction would permit.

### **C.      U.S. Patent No. 9,521,151 (“the ’151 Patent”)**

#### **1.      “optimizing” / “optimize” (Independent Claims 1, 12)**

<b>VMware’s Proposed Construction</b>	<b>Cirba’s Proposed Construction</b>
This term should not be construed or should be given its plain and ordinary meaning.	“achieving a user-desired state”

**a. VMware's Opening Position**

The terms “optimizing” and “optimize,” which appear in the preambles of claims 1 and 12, state the intended purpose of the claims. They need no construction and should be accorded their plain meaning. For example, Merriam-Webster’s defines “optimize” as “make as perfect, effective, or functional as possible.” (“Optimize,” *Merriam – Webster’s Dictionary & Thesaurus* 585 (2014).) The jury does not need an express definition to understand this plain meaning.

These terms also do not appear in isolation. The full phrase in which “optimizing” appears is “optimizing cloud environment through delegated actions,” and the claims themselves illustrate how to achieve this purpose. Claim 1 requires steps relating to optimizing a cloud environment: “gathering” data, “storing” policies, “evaluating” a possible change,” “generating . . . a recommended change,” and “transmitting” and “performing” a change. Claim 12 requires a similarly configured system. Any construction of “optimizing” in this context would do nothing to assist the factfinder in understanding the scope of the claims.

The specification confirms that the plain meaning applies. It explains that optimizing the cloud environment may “us[e] one or more constraint optimization algorithms that . . . may produce a proposed new optimal state.” (’151 patent, 6:62–7:5; *see also id.*, 9:43–48 (“FIG. 6 illustrates a Recommendation Engine 150 that may produce an optimum recommended change to a cloud environment consistent with one or more user-defined policies”), 9:65–10:3 (executing “one or more Optimization Algorithms 156 on the data to identify one or more optimum changes required to take the cloud environment from its current state to a new recommended state.”).)

Nothing in the claims or specification requires that “optimizing” mean “achiev[e] a user-desired state” as Cirba suggests.



**b.      Densify’s Responsive Position**

The ’151 patent is directed to “optimizing” a cloud environment by achieving a “desired state” through use of “policies.” The intrinsic evidence makes clear that “optimizing” in the context of the invention means to “achieve a user-desired state.” For example, the “Brief Summary of the Invention” discusses a “policy-driven method for optimization” where the method “determines whether a cloud environment should be changed to *achieve a more desired state* based on at least one user-defined policy.” ’151 patent, 1:45-57; *see also* ’151 patent at 1:8-11, 1:61-2:11, 3:33-65; 4:6-23; 4:51-61; 4:61-5:18; 5:23-35; 5:38-53; 5:54-67; 6:22-33; 7:8-23; 7:42-8:50; 8:64; 9:11-42; 9:43-65; 9:65-10:16; 10:17-67; 10:67-51; 12:21-53.

The prosecution history supports Densify’s construction. In distinguishing a prior art reference (U.S. Publication No. (US 2013/0073724) to “Parashar et al.” (hereinafter “Parashar”)), patentees argued that Parashar “does not teach monitoring current state of operations of a cloud, setting a *desired state* of operations, and making recommendations to realize the *desired state*—i.e., optimizing the operations of the cloud environment as recited in claim 1.” D.I. 737-3, Ex. K at VMW00100098. Similarly, in a statement relating to the “Allowable Subject Matter,” the Examiner noted that the prior art did not disclose a “system, and method for optimizing a cloud environment . . . [b]ased on the evaluation, generate, a recommended change to the operating conditions of the cloud environment to *achieve at least one desired state* consistent with the at least one user-defined policy; and [t]ransmit the recommended change to the cloud environment for changing the cloud environment from the current state to *the desired state*.” *Id.* at VMW001000109 (emphasis added).

VMware’s proposal that “optimizing” be given its “plain and ordinary meaning” is contrary to the specification’s disclosures and would result in indefiniteness. VMware cites Merriam-Webster’s dictionary definition of “optimize” (“make as perfect, effective, or functional as

possible”) as evidence that the term needs no construction. VMW Op. at 30. But by that definition, “optimizing” the cloud environment of the claim would require making that cloud environment “as perfect *as possible*.” The specification does not provide disclosure sufficient to teach a POSA how to make the cloud environment as perfect “as possible” (an objective standard that is nowhere defined) – it only teaches how to optimize in accordance with a user’s desired state (a subjective standard defined through policies).

### c. VMware’s Reply Position

Cirba ignores that this term describes the *purpose* of the claimed method, as reflected in the combined steps of claim 1. These steps illustrate how to optimize, without deviating from plain meaning or requiring an interpretive gloss for the term. (*See* Menascé Rep. Decl. ¶¶ 11-12.)

Cirba fails to identify intrinsic support for its narrow definition of “achieving a user-desired state.” Cirba does not explain its contention that the specification “only teaches how to optimize in accordance with a user’s desired state (a subjective standard defined through policies),” relying largely on a string cite. (Resp. Br. at 31-32.) Cirba’s position is contrary to the specification’s explanation that “Constraint Engine 155 may be capable of executing one or more Optimization Algorithms 156 on the data to identify one or more optimum changes required to take the cloud environment from its current state to a new recommended state.” (’151 patent, 9:65–10:3.) The Optimization Algorithm, in turn, “may include proprietary algorithms, or published algorithms known to one skilled in the art.” (*Id.*, 10:3–6.) A POSITA would understand how such an algorithm uses policies to recommend optimal changes to the cloud environment, but these recommendations need not be based on a “user-desired” state. (*See* Menascé Rep. Decl. ¶ 13.)

The prosecution history supports VMware’s plain meaning construction. In response to an Office Action, VMware explained that “Claim 1 is directed to ‘optimizing cloud

environment” and “does so by” carrying out the steps of claim 1: gathering data, determining a current state, evaluating using policies whether the cloud environment should be changed to achieve a more desired state, generating a recommended change, and transmitting that change. (D.I. 737-3 (Ex. K) at 164 (VMW00100097).)

**d.      Densify’s Sur-Reply Position**

“The parties agree that the preambles to claims 1 and 12 of the ’151 patent are limiting.”<sup>11</sup> D.I. 737-1, Ex. C. Construing “optimizing” according to its “plain and ordinary meaning” is contrary to the specification’s disclosures. The specification discloses “a system in which a user can...determine the *optimum changes* to bring an environment back to the *desired state* ....” ’151 patent, 1:28-37 (emphases added). VMware’s citation to the Optimization Algorithm as evidence that optimizing means to reach a “recommended state” ignores disclosures that recommendations are “based on the policies representing *user-desired operation*.” *Id.* at 2:2-8. VMware acknowledges this in its briefing. VMW Reply at 44 (“the claim language expressly links ‘the desired state’ to the ‘recommended change’”). A plain and ordinary construction would render the claim indefinite because the patent does not teach how to make cloud environments “as perfect, effective, or functional as possible” (i.e., the ordinary definition of “optimal”).

**2.      “operating conditions” (Independent Claims 1, 12)**

<b>VMware’s Proposed Construction</b>	<b>Cirba’s Proposed Construction</b>
Not indefinite. This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

---

<sup>11</sup> More than simply stating the “purpose” of the claimed method, “the preamble establishes antecedent basis for a term in the claim body” (claim 9, “cloud optimization system”) and “the preamble recites structure or steps underscored as important by the specification” (the specification uses “optimizing” and its variants dozens of times (Title, Abstract, and Summary of the Invention). *Catalina Mktg. Int’l, Inc. v. Coolsavings.com, Inc.*, 289 F.3d 801, 808-09 (Fed. Cir. 2002).

**a. VMware’s Opening Position**

The phrase “operating conditions” in claims 1 and 12 is not indefinite. Per its ordinary meaning, this phrase describes the conditions of the operation of a cloud environment, such as information about and settings of the cloud environment that may be collected, processed, and changed as part of the disclosed methods and systems. This understanding is readily apparent from the specification, which describes exemplary operating conditions for the disclosed inventions as “data from the running infrastructure and optionally the supporting services of the cloud provider,” “cost, performance, configuration, availability and security information about the resources,” “data on the current usage of cloud infrastructure and pricing from the cloud provider,” “current configuration and security settings for all resources comprising the public website,” and “all the virtual instances running the analytics workload, their performance characteristics, and the number of jobs they have processed down to an hourly basis.” (*See* ’151 patent, 3:45–50, 9:5–19, 10:26–39, 11:10–24, 11:57–65.) And Claim 1 recites clear actions involving the operating conditions, such as: “gathering data representing operating conditions of a cloud environment,” “determining a current state of the operating conditions of the cloud environment,” and “generating . . . a recommended change to the operating conditions of the cloud environment.” Finally, the Examiner allowed the claims after an amendment adding “operating conditions” language without raising any issue of indefiniteness. (*See* D.I. 737-3 (Ex. K) at 158 (VMW00100091), 160-161 (VMW00100093-94), 169 (VMW001000107).)

**b. Densify’s Responsive Position**

The ’151 patent specification contains no description of “operating conditions” of a cloud environment. The sole instance of the term “operating conditions” in the specification is in the “Background of the Invention” section. ’151 patent, 1:25-28 (“In order to maintain a desired state for infrastructure, it is necessary to constantly monitor the current state of infrastructure and

regularly tune it to stay within desired *operating conditions*.”). In context, this use of “operating conditions” appears to be synonymous with “state,” *i.e.*, “monitor the current state of infrastructure and regularly tune it to stay within a desired [state].”<sup>12</sup> But if “operating conditions” is construed to be “state,” then the claim would suffer internal redundancies, causing indefinite scope:

gathering data representing [the *state*] of a cloud environment using a monitoring system executing at the computer processor, which is operatively connected to the cloud environment over a network;

determining a current state of the [state] of the cloud environment;

\* \* \*

evaluating, using a policy engine executing at the computer processor, whether the cloud environment should be changed to achieve a more desired state based on the current state of the [state] and the at least one policy set representing user desired operation of the cloud environment and the at least one user-defined policy to authorize a change to the [state] of the cloud environment;

based on the evaluation, generating, using a recommendation engine executing at the computer processor, a recommended change to the [state] of the cloud environment to achieve at least one desired state consistent with the at least one user-defined policy;

A POSA would not be able to reasonably ascertain what qualifies as a “state of the state,” making the scope of the claim ambiguous.

### c. VMware’s Reply Position

Ignoring the plain, unambiguous claim language and the specification’s discussion of actual operating conditions (*see* Op. Br. at 34-35), Cirba asserts that a POSITA would find the term “operating conditions” to be ambiguous. (Resp. Br. at 35-36.) But Cirba cites neither legal authority nor expert support. The term “operating conditions” is a widely used term in the art, and a POSITA would readily understand the scope of this term in the context of “operating conditions of a cloud environment” and the claims. (Menascé Rep. Decl. ¶¶ 14-16.)

---

<sup>12</sup> This construction aligns with extrinsic dictionary evidence which defines “condition” as “a particular *mode of being* of a person or thing; *existing state*; situation with respect to circumstances”; “state of health”; or “fit or requisite state.” *Random House Webster’s Dictionary* 147 (4th ed. 2001) (emphasis added).

Cirba’s argument that the specification “contains no description” focuses only on the verbatim words “operating conditions.” (Resp. Br. at 35.) But “[t]here is no requirement that every term appearing in the claims must be specifically defined or used in the specification.” *Magna Elecs., Inc. v. TRW Auto. Holdings Corp.*, No. 12-654, 2015 WL 11401855, at \*14 (W.D. Mich. Apr. 28, 2015); *see also In re Skvorecz*, 580 F.3d 1262, 1268-69 (Fed. Cir. 2009). Cirba overlooks that the specification describes operating conditions and its use in the claim. (*See, e.g.*, ’151 patent, 3:45-50, 9:5-19, 10:26-39, 11:10-24, 11:57-65; Op. Br. at 34-35.) Cirba further ignores VMware’s prosecution history support for this limitation. (Op. Br. at 34-35.)

Cirba contends that “operating conditions” are the same as another claim term, “state,” allegedly rendering the claim indefinite. (Resp. Br. at 35-36.) But the specification does not equate “operating conditions” with “state,” and Cirba’s strawman argument cannot “constitute evidence of indefiniteness.” *Mycone Dental Supply Co. v. Creative Nail Design, Inc.*, No. 12-00747 RS, 2013 WL 3656379, at \*5 (N.D. Cal. July 12, 2013).

#### **d.      Densify’s Sur-Reply Position**

Unlike the fact patterns in VMware’s cited cases, “operating conditions” *does* appear in the specification, which uses it synonymously with “state.” *Cf. Magna Elecs., Inc. v. TRW Auto. Holdings Corp.*, No. 12-654, 2015 WL 11401855, at \*14 (W.D. Mich. Apr. 28, 2015) (the disputed term “does not appear anywhere in the patents”); *see also In re Skvorecz*, 580 F.3d 1262, 1268 (Fed. Cir. 2009) (“segments” is not defined in the specification). When viewed in light of the claim language and specification, “operating conditions” is indefinite.

### **3.      “the at least one policy set” (Independent Claims 1, 12)**

<b>VMware’s Proposed Construction</b>	<b>Cirba’s Proposed Construction</b>
Not indefinite. The “the at least one policy set representing user desired operation” of Claim 1(d)/12(c) is the “at least one policy representing user desired operation” in Claim	Indefinite

VMware's Proposed Construction	Cirba's Proposed Construction
1(c)/12(b).	

**a. VMware's Opening Position**

The phrase “the at least one policy set” appears in the “evaluating” step of claim 1 (and the “policy engine” element of claim 12). Comparing this “evaluating” step to the “storing” step that precedes it in claim 1, it is readily apparent that “*the at least one policy set* representing user desired operation of the cloud environment” of the “evaluating” step has its antecedent basis in “*at least one policy* representing user desired operation of the cloud environment” in the “storing” step. The mere appearance of the word “set” does not render the claim indefinite, as the plain meaning of “policy set” is one or more policies. This confirms that “the at least one policy set” refers to “at least one policy representing user desired operation” in the “storing step.

**b. Densify's Responsive Position**

The relevant portions of the claim at issue are:

storing, at the memory, *at least one policy* representing user desired operation of the cloud environment, and at least one user-defined policy to authorize a change to the cloud environment;

evaluating, using a policy engine executing at the computer processor, whether the cloud environment should be changed to achieve a more desired state based on the current state of the operating conditions and *the at least one policy set* representing user desired operation of the cloud environment and the at least one user-defined policy to authorize a change to the operating conditions of the cloud environment;

It is clear from the claim language itself that the term “at least one policy set” has no antecedent basis. VMware encourages the Court to construe “the at least one policy set” of the “evaluating” step to be the “at least one policy” of the “storing” step by arguing that “the plain meaning of ‘policy set’ is one or more policies.” VMW Op. at 37. This argument is unavailing for three reasons: (1) the term “set” is repeatedly and consistently used in the specification to include *more than one* thing; (2) claim construction principles forbid simply rewriting the claim language to

resolve an ambiguity; and (3) even if the antecedent basis issue were corrected (by replacing “the” with “an”), that would still not permit a POSA to ascertain, with reasonable certainty, the scope of the claim.

The term “set” is used over a dozen times in the specification. *Every* use refers to multiples of whatever is in the set, including policies: *e.g.*, “set of (user-defined) policies” (’151 patent, 1:53, 67; 2:1-2); “set of instructions” (*id.* at 12:62-63; 13:2-3, 14:20, 15:4, 7, 38); “set of privileges” (*id.* at 7:21-22); “set of operations” (*id.* at 4:1); “set of credentials” (*id.* at 4:33-34). It is improper to ignore the patentees’ clear intent that “set” means “two or more” in favor of VMware’s acontextual assertion that “set” must mean “one or more.” VMW Op. at 37. The specification-defined construction of “policy set” must mean “two or more policies.” This makes VMware’s proposal of simply equating “at least one policy set” with “at least one policy” untenable – the former requires at least two policies while the latter requires only one, making them non-equivalents.<sup>13</sup>

Even if the Court were to correct the antecedent basis by changing “the” to “an”, a POSA would still be unable to ascertain the scope of the claims with reasonable certainty, because the specification does not teach “evaluating...whether the cloud environment should be changed” based on one or more “policy sets” different than the one or more policies stored in memory.

### **c. VMware’s Reply Position**

Cirba’s arguments about antecedent basis and the meaning of the word “set” do not

---

<sup>13</sup> Although it has yet to do so, VMware may urge the Court to simply strike the term “set” from the “evaluating” step. This would be an improper redrafting of claims. *Dickinson & Co. v. C.R. Bard Inc.*, 922 F.2d 792, 799 n.6 (Fed. Cir. 1990) (“Nothing in any precedent permits judicial redrafting of claims.”). Moreover, constructions that result in surplusage (something that would happen if “set” were given no weight) are disfavored. *Texas Instruments Inc. v. United States Int’l Trade Comm’n*, 988 F.2d 1165, 1171 (Fed. Cir. 1993) (rejecting construction that resulted in surplusage). It is also helpful to note that, by all indications, VMware’s use of the term “policy set” was intentional. The term appears in the original draft of the claims submitted for prosecution. VMW00100003-038 at 033, 035. And the term was cited in a response to an Office Action mailed April 14, 2016. D.I. 737-3, Ex. K at VMW00100097.



render the claims indefinite. A POSITA understands that “the at least one policy set representing user desired operation” of claim 1(d)/12(c) refers to the “at least one policy representing user desired operation” in claim 1(c)/12(b). Read in that manner, the claims make sense (Op. Br. at 37), and Cirba has not presented another reasonable construction (Resp. Br. at 36-39).

Instead, Cirba contends that the specification *defines* the term “set” to mean “two or more,” as it refers to a “set of policies,” a “set of instructions,” and a “set of operations,” (Resp. Br. at 38-39.) But Cirba does not explain why the mere use of the term “set” requires that it be “two or more.” A POSITA would find that the only reasonable way to read the claim as a whole is to interpret “at least one policy set” as “at least one policy.” (*See* Menascé Rep. Decl. ¶¶ 17-18.) Moreover, a POSITA was familiar with the general, mathematical principle that a “set of objects” may have one or more objects. (*See id.* ¶ 19.)

Cirba asserts that striking the word “set” from the claims would be improper. (Resp. Br. at 38-39 & n.13.) Although striking the word is unnecessary because the claim scope is clear, the Court nevertheless may do so if it finds that “set” is a clerical error. *See Novo Indus., L.P. v. Micro Molds Corp.*, 350 F.3d 1348, 1357 (Fed. Cir. 2003) (“courts can continue to correct obvious minor typographical and clerical errors in patents”). Here, striking “set” would be consistent with VMware’s construction, the only reasonable one.

#### **d.      Densify’s Sur-Reply Position**

VMware agrees that this term contains an error but insists that the error is the inclusion of the word “set” and not the lack of antecedent basis. VMware proposes the correction “the at least one policy ~~set~~,” while dismissing the equally viable correction “~~the~~ [an] at least one policy set.” VMW Reply at 39. VMware and its expert Dr. Menasce contend that their construction is “the only reasonable way” to read the claim (*id.*), but this argument ignores the fact that, in addition to teaching single “policies,” the specification teaches “policy *sets*” containing multiple policies

(Densify Opp. at 38), thereby leaving open the possibility that the “evaluating” step requires using a multi-policy “policy set.”

#### 4. “objective” (Dependent Claims 3, 4, 14, 15)

VMware’s Proposed Construction	Cirba’s Proposed Construction
This term should not be construed or should be given its plain and ordinary meaning.	“a desired state”

##### a. VMware’s Opening Position

“Objective” is a common English word that does not require construction. An objective is a goal. (*See, e.g., “Objective,” Merriam-Webster Dictionary & Thesaurus*, 574 (2014) (defining “objective” as “an aim, goal, or end of action”).) The claim’s use of this ordinary meaning is clear from claims 3 and 4, which state, respectively, that “the at least one user-defined policy defines at least one objective” and “the at least one objective is at least one of the following: lower cost; increased availability; better performance; and higher security.” The specification also supports giving “objective” its ordinary meaning in English by referring to several goals achieved by the claimed inventions. (*See* ’151 patent, 1:8–11 (“securely automating recommendations to optimize the cost, availability, performance, and security of cloud infrastructure”), 6:10–13 (“automated orchestration of cloud environments to achieve defined business goals”), 9:30–36 (example policy for “cost-optimization”).)

Cirba’s proposed construction—“desired state”—is a different term appearing in claims 1 and 12. The two terms should not be conflated. *See, e.g., Helmsderfer v. Bobrick Washroom Equip., Inc.*, 527 F.3d 1379, 1382 (Fed. Cir. 2008) (“different claim terms are presumed to have different meanings”). Cirba’s confusing construction apparently seeks to manufacture indefiniteness where none exists. (*See infra* §§ III.E, III.H (Cirba contending “the desired state” and “at least one desired state” are indefinite).)

**b.      Densify’s Responsive Position**

Claims 3 and 14 state “the at least one user-defined policy defines at least one objective,” and claims 4 and 15 state “wherein the at least one objective is at least one of the following: lower cost; increased availability; better performance; and higher security.” ’151 patent, 16:59-65. The specification explains that “[a] *policy* represents a *desired state* for a cloud environment.... The desired state can span a plurality of objectives, such as cost, performance, availability and security.” *Id.* at 3:11-18. In a classic case of the transitive property (e.g., “if A = B, and B = C, then A = C”), the specification defines an “objective” as “a desired state.”

**c.      VMware’s Reply Position**

Cirba does not dispute that VMware’s Opening Brief applies the plain meaning of this common English word: an objective is a “goal.” Nor does Cirba dispute that the several examples of the term “objective” in claim 4 use this plain meaning. (Resp. Br. at 41.) Without any authority or evidence, Cirba asserts that “[i]n a classic case of the transitive property, . . . the specification defines an ‘objective’ as ‘a desired state.’” (*Id.*) Cirba ignores that the specification has no express lexicography or disclaimer for the word “objective.” *See Toshiba*, 681 F.3d at 1369. Cirba’s “transitive property” argument also offers no reason to conflate three distinct claim terms: objective, policy, and desired state.

**d.      Densify’s Sur-Reply Position**

Claims 3 and 4 use the term “objective” consistently – that is, a “desired state,” as taught in the specification. ’151 patent, 3:11-18 (“A *policy* represents a *desired state* for a cloud environment.... The desired state can span a plurality of *objectives*, such as cost, performance, availability and security.”).

5. “the desired state” (Independent Claims 1, 12)

VMware’s Proposed Construction	Cirba’s Proposed Construction
Not indefinite. This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

a. VMware’s Opening Position

Cirba contends that the term “the desired state” in claim 1’s “transmitting” step (and claim 12’s “recommendation engine” limitation) is indefinite. But claim 1 provides a clear antecedent basis for “the desired state” in the preceding “generating” step: “generating . . . a recommended change to the operating conditions of the cloud environment to achieve at least one desired state consistent with the at least one user-defined policy.” Moreover, the specification supports applying the plain meaning to this phrase, as used in the claims. (*See, e.g.*, ’151 patent, 1:54–60 (“The method determines whether a cloud environment should be changed to achieve a more desired state based on at least one user-defined policy.”), 2:2–11 (“The system may also comprise a recommendation engine that produces a recommended change to the cloud environment to achieve at least one desired state consistent with the at least one user-defined policy”), 6:1–10 (“a recommendation engine capable of proposing a change to achieve a desired state based on a known policy”).) The term is not indefinite and needs no express construction.

b. Densify’s Responsive Position

The claim term “*the* desired state” is *per se* indefinite because it has two antecedent bases.

Claim 1 recites, in relevant part:

evaluating, using a policy engine executing at the computer processor, whether the cloud environment should be changed to achieve ***a more desired state*** based on the current state of the operating conditions and the at least one policy set representing user desired operation of the cloud environment and the at least one user-defined policy to authorize a change to the operating conditions of the cloud environment;

based on the evaluation, generating, using a recommendation engine executing at the computer processor, a recommended change to the operating conditions of the cloud

environment to achieve *at least one desired state* consistent with the at least one user-defined policy;

transmitting the recommended change to the cloud environment for changing the cloud environment from the current state to *the desired state*;

'151 patent at 16:30–45 (emphasis added). The “evaluating” step recites “a more desired state,” while the “generating” step recites a different “at least one desired state.”<sup>14</sup> The “transmitting” step simply recites “*the* desired state,” without making clear which of the prior two desired states is being referred to.<sup>15</sup>

VMware argues that the “plain and ordinary” meaning of “the desired state” is the “at least one desired state” of the immediately preceding “generating” step but offers no reasonable argument in support. VMware’s specification citations actually underscore the ambiguity. Each uses non-specific language that mirrors the claim language without resolving *which* desired state is claimed in the “transmitting” step. *See, e.g.*, 1:54-60 (“changed to achieve *a more desired state*”), 2:2-11 (“change to the cloud environment to achieve *at least one desired state*”), and 6:1-10 (“a change to achieve *a desired state*”).

### c. VMware’s Reply Position

Cirba asserts that the term “the desired state” in the claims is “*per se* indefinite because it has two antecedent bases.” (Resp. Br. at 43-44.) Cirba cites no authority for this proposition, and the claims’ structure shows that “the desired state” term does not have two antecedent bases.

In the claims, “the desired state” of the “transmitting” step is the “at least one desired

---

<sup>14</sup> VMware does not argue for, and the plain language of the claim does not support, equating “more desired state” with “at least one desired state.” In fact, “a *more desired state*” is not necessarily a *desired state*. For example, if the temperature in a home measures 100 degrees Fahrenheit on a hot summer day, a “more desired state” may be 98 degrees, but that does not necessarily mean that 98 degrees itself is actually *desired*.

<sup>15</sup> VMware does not argue that, and there is no support for, the “more desired state” is the same as the “at least one desired state.”

state” of the previous “generating” step, as the claim language expressly links “the desired state” to the “recommended change.” (’151 patent, cl. 1 (“generating . . . a recommended change . . . to achieve at least one desired state”; “transmitting the recommended change . . . for changing . . . to the desired state”); *see also* Menascé Rep. Decl. ¶¶ 20-21.) A POSITA would understand that this construction is the only reasonable interpretation of the claims. (*Id.*)

Cirba offers no reason why a POSITA would consider the prior “evaluating” step—in which the method considers a possible “more desired state”—to be the antecedent basis of “the desired state” of the “transmitting” step. Nor does the specification confuse the antecedent basis for the desired state of the “transmitting” step, as Cirba asserts. (Resp. Br. at 43-44.) To the contrary, Cirba’s cited portions of the specification confirm the invention “evaluates whether the cloud environment should be changed to achieve a more desired state” and then generates a recommended change “to achieve at least one desired state.” (’151 patent, 2:2–11; *see also id.*, 1:54–60.) Nothing in the specification requires departing from the plain structure of the claim language in which the “transmitting” step naturally follows the “generating” step. *See Mantech Envtl. Corp. v. Hudson Envtl. Servs., Inc.*, 152 F.3d 1368, 1376 (Fed. Cir. 1998).

#### **d.      Densify’s Sur-Reply Position**

The “at least one desired state” term of the “generating” step must be “consistent with the at least one user-defined policy,” which is associated with the “more desired state” of the “evaluating” step. Thus, VMware’s assumption that the claim language “expressly links ‘the desire state’ to the ‘recommended change’” is not dispositive – the claim language also links ‘the desired state’ to the ‘more desired state’ element. VMware’s citation to *Mantech Envtl. Corp. v. Hudson Envtl. Servs., Inc.*, 152 F.3d 1368, 1375 (Fed. Cir. 1998), which concerns a claim where the “steps must be performed in order,” is inapplicable. VMware does not contend that the steps here must be performed sequentially (*cf.* ’049 patent, D.I. 737-1, Ex. A) Thus, “the desired state”

may refer to either antecedent, making the term indefinite.

**6. “user-defined threshold” (Dependent Claims 2, 13)**

<b>VMware’s Proposed Construction</b>	<b>Cirba’s Proposed Construction</b>
Not indefinite. This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

**a. VMware’s Opening Position**

The phrase “user-defined threshold,” which appears in the limitation “determining whether a user-defined threshold is met,” needs no construction in the context of claim 2 and 13 and the specification. The preceding limitation of claim 2 is “monitoring the cloud environment by the monitoring system for deviation from the at least one user-defined policy.” The disclosed inventions may “monitor the cloud infrastructure resources for deviation from the policy, and propose corrective action.” (’151 patent, 12:51–53.) In light of the “monitoring” step of claim 2, the deviation in claim 2 is measured against a “user-defined threshold.”

The specification explains that “[w]ithin a policy may be one or more explicit or implied trigger conditions. The trigger condition may define when a cloud environment has drifted from a desired state such that it needs to be modified to bring it back within the policy.” (*Id.*, 3:22–26.) The patent then gives an example in which an organization may set a trigger condition for search clusters to be managed to 99.9% availability, which is a “user-defined threshold.” (*Id.*, 3:26–32.) In another example, the patent describes a threshold for capacity reservations in a cloud environment, where purchase of reserve capacity must achieve no less than a 40% cost benefit, a “user-defined threshold.” (*Id.*, 7:55–65.) And in yet another embodiment, the specification discloses “rightsiz[ing] a workload upon identifying that a workload has been provisioned with excess or insufficient resources” (*id.*, 12:37–40), which may be based on a resource threshold specifying “excess” or “insufficient.”

This intrinsic record shows that claims 2 and 13 use the phrase “user-defined threshold” in its ordinary sense. (*See, e.g., “User-defined,” Webster’s New World Computer Dictionary*, 387 (10th ed. 2003) (defining “user-defined” as “[s]elected or chosen by the user of the computer system”); “Threshold,” *Collins Dictionary*, 1702 (11th ed. 2011) (defining “threshold” as “a level or point at which something would happen, would cease to happen, or would take effect, become true, etc.”).) No further construction is needed, and the term is not indefinite.

**b.      Densify’s Responsive Position**

The term “user-defined threshold” is indefinite for at least two reasons: (1) it is nowhere discussed in the specification; and (2) the claim language does not state what is being measured against the “threshold” to determine if it is met.

VMware points to portions of the specification that discuss monitoring (e.g., “cloud infrastructure resources” for “deviation” from a policy), but none provide definiteness to the scope of the invention. Claim 1, from which claim 2 depends and to which VMware points as providing context for the claimed “threshold,” does not limit what the “threshold” can be – in other words, because it is not recited as being related to anything, it can be *anything*. As long as a user defines some threshold – *any* threshold – that element is met. VMware’s “plain and ordinary” definition reinforces the breadth of the limitation.

Claim 2 also does not recite *what* the “user-defined threshold” is being measured against to determine if it is met. A POSA would necessarily wonder if it is the “deviation” of the first element or something else. Because the “threshold” could be anything, it need not be the metric against which a deviation may be measured.

**c.      VMware’s Reply Position**

Cirba’s assertion that the term “user-defined threshold” is indefinite because it is “nowhere discussed in the specification” (Resp. Br. at 46-47) both is untrue and ignores its plain



meaning in common English. (Op. Br. at 45-46.) The specification supports the term with numerous examples of user-defined thresholds. (*Id.*; '151 patent, 3:22–32, 7:55–65, 12:37–40; *see also* Menascé Rep. Decl. ¶ 24.) And “[t]here is no requirement that [this term] must be specifically defined or used in the specification.” *Magna Elecs.*, 2015 WL 11401855, at \*14.

Cirba complains that adopting a plain and ordinary meaning of “user-defined threshold” would “reinforce[] [its] breadth.” (Resp. Br. at 46-47.) But breadth alone cannot make a claim indefinite. *See BASF Corp. v. Johnson Matthey Inc.*, 875 F.3d 1360, 1367 (Fed. Cir. 2017); *Alarm.com, Inc. v. SecureNet Techs. LLC*, 345 F. Supp. 3d 544, 549-50 (D. Del. 2018).

Cirba contends that claim 2 “does not recite what the ‘user-defined threshold’ is being measured against to determine if it is met.” (Resp. Br. at 47.) Cirba fails to read this term in “the context of the claim as a whole.” *Kyocera Wireless Corp. v. I.T.C.*, 545 F.3d 1340, 1347 (Fed. Cir. 2008). A POSITA would understand that claim 2’s method measures the “threshold” against the deviation. (Menascé Rep. Decl. ¶ 25; Op. Br. at 45-46.) VMware’s construction is reasonable, and Cirba neither suggests an alternative nor identifies a fatal ambiguity.

#### **d.      Densify’s Sur-Reply Position**

VMware’s citations to the specification as to what a “user-defined threshold” might be are exemplary and not limiting. VMW Reply at 46-47. Unbounded, the recited “threshold” could be anything (a condition VMware does not contest). Although it is possible that the threshold may refer to the “deviation” recited in the prior element, nothing requires that it do so, thereby leaving a POSA unable to reasonably ascertain whether or when a “user-defined threshold is met.”

#### **7.      “at least one recommended action” (Dependent Claims 2, 5, 6)**

<b>VMware’s Proposed Construction</b>	<b>Cirba’s Proposed Construction</b>
Not indefinite. This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

**a. VMware’s Opening Position**

The phrase “at least one recommended action” in dependent claims 2, 5, and 6 is ordinary English and requires no construction. The phrase appears in the context of Claim 2, described above in Section III.F, which provides for monitoring the cloud environment for deviations from a user-defined policy that meet a user-defined threshold. If such a deviation meets the threshold, the claim further requires making “at least one recommended action” for changing the cloud environment. Nothing about this is unclear or requires further construction.

The patent explains how this works in an illustrative embodiment. “Cloud Optimization System 10 may understand the policy, monitor the cloud infrastructure resources for deviation from the policy, and propose corrective action.” (’151 patent, 12:51–53.) The specification also provides examples, such as “rightsiz[ing] a workload upon identifying that a workload has been provisioned with excess or insufficient resources” and “defin[ing] a corrective action.” (*Id.*, 12:37–46.) The term is neither indefinite nor requires construction.

**b. Densify’s Responsive Position**

Claims 2, 5, and 6 recite “a recommended action for changing the cloud environment” to achieve the desired state recited in claim 1, but nothing in the claim recites who or what determines the “recommended action.”

Claim 2 recites in relevant part:

The method of claim 1, further comprising the steps of:  
 monitoring the cloud environment by the monitoring system for deviation from the at least one user-defined policy;  
 determining whether a user-defined threshold is met;  
 making *at least one recommended action* for changing the cloud environment to achieve the at least one desired state.

A POSA would not be able to ascertain with reasonable certainty how the recommended action is determined. Unlike claim 1, claim 2 does not recite how the recommended action is generated or what variables are used in generating it. Nothing in the claim language links the recommended action to the “deviation” recited in the “monitoring” element or the “user-defined threshold” in the “determining” element. Thus, the nature of the recommended action, and how and when to “make” it, are wholly unknown.<sup>16</sup>

**c. VMware’s Reply Position**

Cirba ignores the plain English meaning of these words and the exemplary support in the specification. (Resp. Br. at 49). The patent describes an embodiment that “monitor[s] the cloud infrastructure resources for deviation from the policy, and propose[s] corrective action.” (’151 patent, 12:51-53.) In view of this, a POSITA would understand that the claimed method requires “at least one recommended action” for changing the cloud environment if a monitored deviation meets a user-defined threshold. (Menascé Rep. Decl. ¶¶ 26-27; Op. Br. at 48.)

Cirba complains that “[a] POSA would not be able to ascertain with reasonable certainty how the recommended action is determined” and “claim 2 does not recite how the recommended action is generated or what variables are used.” (Resp. Br. at 49). But Cirba cites no authority finding that such complaints render a claim indefinite. A POSITA would have no difficulty understanding this term. (*See also* Menascé Rep. Decl. ¶¶ 26-27.)

**d. Densify’s Sur-Reply Position**

VMware recognizes that “making at least one recommended action” may happen only “if

---

<sup>16</sup> Claim 5 is further indefinite for the added reason that the *required* step of “making [the] at least one recommended action” recited in claim 2 (from which claim 5 depends) is demoted to an optional step. ’151 patent, cl. 5 (“wherein the at least one recommended action, *if executed*”). Claim 6 is further indefinite for the added reason that it appears to claim a new “at least one recommended action,” which is not recited as having an antecedent basis.

a monitored deviation meets a user-defined threshold.” VMW Reply at 49. For the reasons set forth in Densify’s briefing, “user-defined threshold” is indefinite, thereby causing the “at least one recommended action” to be disconnected from anything that would inform a POSA how the recommended action is generated or what variables are used to generate it. That a POSA has the option of interpreting it as being related to the “deviation” element does not make the term definite because there is no reasonable certainty that that option is correct.

**8. “a recommended change to the operating conditions of the cloud environment to achieve at least one desired state consistent with the at least one user-defined policy” (Independent Claims 1, 12)**

<b>VMware’s Proposed Construction</b>	<b>Cirba’s Proposed Construction</b>
Not indefinite. This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

**a. VMware’s Opening Position**

This phrase appears in claims 1 and 12 and requires no construction, as its plain meaning will be understandable to the jury. The specification confirms that the phrase is used consistent with its ordinary meaning. The specification describes in an example “a system that collects and stores data on the current state of a cloud-based environment having a policy engine capable of interpreting policies describing the desired states for one or more environments, a recommendation engine that can propose changes to an environment to make it consistent with the policies, and execution capability for executing these proposed changes upon receipt from outside the system of the necessary security authorization.” (’151 patent, 5:54–67.)

Figure 6 illustrates the claimed “recommended change” using a “recommendation engine.” (*Id.*, 9:11–48, FIG. 6.) As an example embodiment of claim 1, Figure 7 and associated text in the specification also describe a cloud instance “reservation purchase policy” used to generate recommended changes to the cloud environment. (*Id.*, 9:43–10:67, FIG. 7.) Data on

the current usage of the cloud infrastructure and pricing, as well as a purchase history, are collected. (*Id.*, 10:26–31.) The reservation purchase policy then is evaluated against the data, looking for cloud usage optimization opportunities. (*Id.*, 10:31–39.) A user-defined policy may include automated approval for a purchase of a trusted system for less than \$5,000. (*Id.*, 10:58–61.) This example identifies a recommended change based on this evaluation: “Make a \$180,000 purchase of 10 big instance types in region US-East for heavy use . . .” (*Id.*, 10:34–45.) The patent provides ample support for this phrase, which is not indefinite.

### **b.      Densify’s Responsive Position**

This term’s indefiniteness is rooted in the claim language’s requirement that a “desired state” be “consistent” with a “user-defined policy,” to “authorize a change to the cloud environment.” The patent discloses changing the cloud environment to achieve a desired state consistent with the policy representing *user desired operation* of the cloud environment, not a policy that provides *authorization* to change the cloud environment.

Claim 1 recites, in relevant part:

storing, at the memory, *at least one policy representing user desired operation* of the cloud environment, and *at least one user-defined policy to authorize a change to the cloud environment*;

\* \* \*

based on the evaluation, generating, using a recommendation engine executing at the computer processor, a recommended change to the operating conditions of the cloud environment to achieve at least one *desired state consistent with the at least one user-defined policy*;

The claim requires at least two policies be stored in memory: a “policy representing user desired operation” and a “user-defined policy to authorize a change to the cloud environment.” Critically, the two policies are *not* required to be related to one another, meaning the “user-defined policy to authorize a change” could be directed at *anything*, including changes that have nothing to do with the “user desired operation” policy. Thus, when the “generating” step recites generating

“a recommended change to the operating conditions . . . to achieve at least one desired state,” that desired state should be consistent with the policy representing user desired operation, not the (potentially unrelated) user-defined policy to authorize (some non-specified) change.

**c. VMware’s Reply Position**

Cirba’s brief confirms that this claim term is not indefinite. Agreeing with VMware, Cirba correctly notes that the recommended change to achieve the desired state of claim 1 must be “consistent” with the “user-defined policy.” (Resp. Br. at 51-52.) Cirba identifies nothing ambiguous or uncertain about this claim language.

Instead, Cirba complains that the claim language allegedly is insufficiently tied to the specification’s discussion of policies. Per Cirba, the limitation should have been written to require that the recommended change be consistent with the “policy representing user desired operation of the cloud environment.” (*Id.* at 51.) Cirba believes that this alternate claim would have found greater support in the specification.

Even if true, this would not demonstrate indefiniteness. Cirba’s disguised written description attack would fail in any event, as it ignores the specification’s teaching that “Recommendation Engine 150 . . . may produce an optimum recommended change to a cloud environment consistent with one or more user-defined policies.” (’151 patent, 9:45-48.) The specification also refers to “user-defined” policies that “authorize a change to the cloud environment” as spelled out in the claims. (*Id.*, 1:66-2:11.) (See Menascé Rep. Decl. ¶¶ 28-30.)

**d. Densify’s Sur-Reply Position**

The claim language requires that the recited “at least one desired state” be consistent with the “user-defined policy,” which (as its antecedent in the “evaluating” step indicates) is a policy to *authorize*. The specification does not teach how to make a “desired state” consistent with a policy to authorize – it teaches how to make a “desired state” consistent with a “user desired

operation.” ’151 patent, 5:54-63. As claimed, the term lacks written description and is indefinite.

**D. U.S. Patent No. 10,069,752 (“’752 Patent”)**

- 1. “interfacing with the distributed computer systems at a remote resource allocation module that is located outside of the distributed computer systems” (Claims 1, 9)**

<b>VMware’s Proposed Construction</b>	<b>Densify’s Proposed Construction</b>
This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

**a. VMware’s Opening Position**

This step does not render claims 1 and 9 indefinite and does not require construction.

The language of Claims 1 and 9 is straightforward. *See Cox Commc’ns, Inc. v. Sprint Commc’ns Co.*, 838 F.3d 1224, 1231-32 (Fed. Cir. 2016) (“As *Nautilus* [572 U.S. at 909] instructs, the dispositive question in an indefiniteness inquiry is whether the ‘claims,’ not particular claim terms, ‘read in light of the specification delineating the patent, and the prosecution history, fail to inform, with reasonable certainty, those skilled in the art about the scope of the invention.’”).

Claims 1 and 9 are written from the perspective of a computer performing remote resource allocation analyses and employing a “remote resource allocation module.” Per this “interfacing” step, the remote resource allocation module is located outside of and interfaces with the distributed computer systems that are being analyzed. A requirement for this interfacing step is “receiving requests for access to the remote resource allocation module using authorization information.” None of this language injects any uncertainty. The remaining steps of claims 1 and 9, after interfacing, recite steps taken to carry out the remote resource allocation analyses and transmit the results to the distributed computer systems.

The specification illustrates the “interfacing” step. Figure 1 shows an embodiment with a distributed computer system and a remote resource allocation module. (’752 patent, 2:43-44, FIG. 1.) This allocation module in Figure 1 is located on a remote server, not on one of the

computers in the distributed computer system, and may establish an interface to the distributed computer system over a network. (*Id.*, 3:46-67 (“[T]he computer network system includes a network 102, distributed computer systems F-1, F-2 . . . F-Y and a remote server 104 with a remote resource allocation module 106. . . . [T]he distributed computer systems are able to communicate with the remote server via the network.”); *see also id.*, 8:26-38, 9:40-10: 11.)

### **b.      Densify’s Responsive Position**

This term is indefinite for two independent reasons: it employs an improper prepositional phrase, and the claim is ambiguous as to *what* is doing the interfacing.

The term “interfacing with...at” does not have any coherent meaning because, in the networking context, the verb “interface” cannot be used with the preposition “at.”<sup>17</sup> Madisetti Dec. ¶ 21. “Interface” simply means “to connect.” *Random House Webster’s Computer & Internet Dictionary* 281 (3d ed. 1999) (“[T]wo devices that can transmit data between each other are said to *interface with each other*.”); Madisetti Dec. ¶ 21. Because a network connection allows devices to communicate from *different* locations, there is no place *at* which they interface.

The specification does not explain what it means for network components to interface “at” a location. The specification uses “interface” as a verb six times (’752 patent at 1:64-66; 2:20-23; 8:17-19; 9:40-45; 12:44-47; 14:49-50), but none of those uses involve objects interfacing “at” a location. *E.g., id.* at 8:17-19 (“The snapshot creation unit interfaces with other components of the

---

<sup>17</sup> In a *non-networking* context, one can imagine a scenario where two people interface with each other “at” a particular location. For example, if Alice and Bob meet in Times Square, it could be said that Alice *interfaced* with Bob *at* Times Square. There is no applicable analogy in the digital- or network- space. If Alice sends Bob a text message, their communication takes place using ephemeral signals – it would not make any sense to say that Alice interfaced with Bob “at” any particular location on that network.



management computer 204 to obtain the information needed to generate the snapshot.”).<sup>18</sup> Absent guidance from the claim or specification, a POSA cannot ascertain the claim’s locational “at” requirement.<sup>19</sup> This renders the claim indefinite. *See Interval Licensing*, 766 F.3d at 1371.

The “interfacing” term is also indefinite because there is no way to discern who or what is doing the interfacing. VMware contends that the claims “are written from the perspective of a computer performing remote resource allocation analyses and employing a ‘remote resource allocation module.’” VMW Op. at 53. But that construction (which was offered without citation) is not supported by the claim language, which does not say *anything* about a computer.<sup>20</sup> VMware’s construction is also inconsistent with the patent’s dependent claims, which state that the method steps are carried out “at the distributed computer systems.” ’752 patent, cl. 5. That language makes clear that the claim steps are not performed by the remote resource allocation module (or its host computer), since the module is “located *outside* of the distributed computer systems.” *Id.*, cl. 1; *see also* Madisetti Dec. ¶ 25; *Phillips*, 415 F.3d at 1314-15 (“Other claims of the patent in question ... can be valuable sources of enlightenment as to the meaning of a claim term.”).

Redrafting the claim to ignore the “at” requirement is legally improper. *Dickinson*, 922 F.2d at 799 n.6 (“Nothing in any precedent permits judicial redrafting of claims.”). Moreover,

---

<sup>18</sup> Unasserted claim 17 also supports Densify’s position; it discloses a processor configured to “interface with the distributed computer systems *to* receive requests for access ....” *Id.*, cl. 17 (emphasis added).

<sup>19</sup> “At” could mean that the person or entity practicing the claim is required to be *physically located* “at a remote resource allocation module.” Alternatively, “at” could mean that the interface must be *initiated* by the entity practicing the claim, rather than by the remote resource allocation module. However, neither possibility finds support in the specification, nor do they make sense when considered against the intended purpose of the claim.

<sup>20</sup> Plaintiff’s “from the perspective of a computer” construction is also nonsensical, since it would require the implementing computer to interface *at itself*.

failing to give meaning to the “at” limitation would result in surplusage, which is disfavored. *See Texas Instruments*, 988 F.2d at 1171 (rejecting construction that resulted in surplusage).

VMware’s “plain and ordinary” construction is inconsistent with the specification, which identifies several instances in “interfacing” is performed by distributed computer systems, rather than by a computer with a remote resource allocation module. For example, Fig. 7, which contains “a flow diagram of a method ... in accordance with an embodiment of the invention,” states that that the method includes “interfac[ing] with a remote resource allocation module from the computer network facility ....” ’752 patent, Fig. 7; 2:58-60; *see also* 2:20-23 (distributed computer system is “configured to interface with a remote resource allocation module”); 9:40-45 (“The remote processing interface unit [of a distributed computer system] operates to interface with the remote resource allocation module 106 in the remote server 104 ...”). VMware’s construction therefore excludes a disclosed embodiment. *CSS Tech., Inc. v. Panduit Corp.*, 778 F. App’x 947, 950 (Fed. Cir. 2019) (“[A]n interpretation which excludes a disclosed embodiment from the scope of the claim is rarely, if ever, correct.”) (internal citations omitted)).<sup>21</sup>

### **c. VMware’s Reply Position**

Cirba advances two grounds for indefiniteness of the “interfacing” step: (1) the common English word “at” is allegedly confusing; and (2) “what is doing the interfacing” is allegedly ambiguous. (Resp. Br. at 54-57.) Neither survives scrutiny.

From the intrinsic record and the plain meaning of the word “at,” a POSITA would understand that interfacing occurs at the location of the remote resource allocation module. The

---

<sup>21</sup> VMware’s construction is also inconsistent with its statement that the claim’s functions “are carried out by an access interface unit of the remote resource allocation module.” Op. Br. at 19. Moreover, it does not make sense to say that the claims are written from the perspective of an access interface unit, both because that that would exclude the embodiments where the interfacing is carried out by distributed computer systems, and because it is inconsistent with the “performing remote resource allocation analyses” limitation, which is *not* performed by an access interface unit.

patent discloses such a module “*physically located outside of the distributed computer systems* . . . and thus, not part of any of the distributed computer systems.” (’752 patent, 12:28-41; Op. Br. at 53-54.) A POSITA thus would understand that “at” in the “interfacing” step refers to the physical or network location associated with the remote resource allocation module located outside the distributed computer systems. (See Menascé Rep. Decl. ¶¶ 31-33.) Although Cirba complains that “a network connection allows devices to communicate from different locations” (Resp. Br. at 54-55), that does not make the claim uncertain. Wherever the distributed computer systems are located, they may communicate with the remote resource allocation module.

The patent illustrates this concept. The two-headed arrow between elements 102 and 104 in Figure 1 depicts “interfacing with distributed computer systems at a remote resource allocation module” with the connection occurring through a network. Figure 6 also provides an example: an Access Interface Unit 602 that is the location of the interface in the remote resource allocation module. *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364 (Fed. Cir. 2014), is not to the contrary (Resp. Br. at 55), as the ’752 patent does not disclose different meanings for “at.”

Cirba’s argument regarding “who or what is doing the interfacing” also fails. (*Id.*) The claims indicate that the remote resource allocation module and the distributed computer systems are interfacing. As discussed above, that is consistent with the specification and also Cirba’s definition of “interfacing” as “connecting,” (*id.* at 55-56; Op. Br. at 53-54). A skilled artisan would have no difficulty understanding this limitation. (Menascé Rep. Dec. ¶ 34.)<sup>22</sup>

Contrary to Cirba (Resp. Br. at 55-56), dependent claim 5 is consistent with VMware’s interpretation. Claim 5 adds the step of performing *local* resource allocation analysis *at* the

---

<sup>22</sup> Cirba is wrong in asserting that the claims do “not say anything about a computer” (Resp. Br. at 55-56). The “remote resource allocation module” is either computing software or hardware. Claim 9 also expressly identifies “a computer system” performing the claimed steps.

distributed computer systems to claim 1. Nothing in claim 5 requires that its added step take place at the same location as claim 1's *remote* analysis steps. Finally, Cirba's specification citations describing elements other than the remote resource allocation module "interfacing" do not render the claims indefinite. (*Id.* at 55-57.) Different disclosed elements may "interface," even if the claims do not cover those examples. *See Pacing Techs., LLC v. Garmin Int'l, Inc.*, 778 F.3d 1021, 1026 (Fed. Cir. 2015) ("every claim does not need to cover every embodiment").

**d.      Densify's Sur-Reply Position**

Not every preposition can be used with every verb. One can cook *with* a friend, but not *at* a friend; or one can sail *on* the high seas, but not *at* the high seas. Likewise, a system may interface *with* a remote resource allocation module (RRAM), but it is nonsensical to interface *at* a RRAM. VMware's citations to the intrinsic evidence do not support its position of non-indefiniteness. Figure 1 does not explain what it means to interface "at" a location – if it shows interfacing at all, it's that interfacing takes place *via* a network connection. Figure 6 depicts an "access interface unit," but "interface" is used as a noun, not a verb. And while it is true that an interface *unit* can exist *at* the RRAM, that provides no insight into what it means *to interface ... at* a RRAM.

When it comes to who or what performs the interfacing, VMware's briefing is inconsistent. In its opening brief, VMware first claimed that the steps are implemented by "a *computer* performing remote resource allocation analyses." VMW Op. at 53 (emphasis added). Later in the brief, it said the steps are carried out by "an *access interface unit* of the [RRAM]." *Id.* at 62 (emphasis added). In its Reply, VMware suggests the "interfacing" step is carried out by the *RRAM* itself.<sup>23</sup> VMW Reply at 57. VMware's confused positioning reflects the claim's

---

<sup>23</sup> VMware's technology tutorial suggests a fourth option: the interfacing is carried out by "a *network* such as the Internet." D.I. 771-1 at 40 (emphasis added).

indefiniteness.<sup>24</sup>

## 2. “remote resource allocation module” (Claims 1, 9)

VMware’s Proposed Construction	Densify’s Proposed Construction
This term should be given its plain and ordinary meaning. “Remote resource allocation module” is not a means-plus-function claim term subject to 35 U.S.C. § 112(f) but complies if it is.	Indefinite  “remote resource allocation module” is a means-plus function claim term subject to § 112(f) with insufficient structure.

### a. VMware’s Opening Position

#### § 112(f) does not apply

“Remote resource allocation module” is not a means-plus-function term. The term does not use the phrase “means,” and Cirba has not “overcome the presumption that § 112[(f)] does not apply to claim terms not including the word ‘means.’” *M2M Sols. LLC v. Sierra Wireless Am., Inc.*, No. 12-30-RGA, 2016 WL 1298961, at \*2, 5-6 (D. Del. Mar. 31, 2016 (finding “memory module” and “processing module” are not means-plus-function terms); *Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1008 (Fed. Cir. 2018) (“[T]hat the disputed limitations incorporate functional language does not automatically convert the words into means for performing such functions.”). The remote resource allocation module has “a known structural meaning, or recites either a known or generic term with sufficient description of its operation,” so “the presumption against means-plus-function claiming remains intact.” *M2M Sols. LLC v. Sierra Wireless Am., Inc.*, No. 12-30-RGA, 2015 WL 5826816, at \*3-4 (D. Del. Oct. 2, 2015) (emphasis omitted).

“Resource allocation modules” were well-known. (Menascé Decl. ¶ 10.) The patent informs a person of ordinary skill in the art (“POSITA”) of the structural element “remote resource allocation module” by invoking the POSITA’s knowledge of resource allocation

<sup>24</sup> The specification discloses embodiments where the “interfacing” is performed by the *distributed computer systems* – yet a fifth option that VMware’s own suggested constructions do not cover.

modules and remote servers. “Resource allocation” is a well-known process, accomplished by software and hardware, for “dividing available resources in a system between jobs.” (“Resource allocation,” *Dictionary of Computing* 285 (6th ed. 2010); Menascé Decl. ¶ 11.) A POSITA was familiar with the design, operation, and use of conventional resource allocation modules, including VMware’s commercially-available DRS installed on a vCenter Server. (’752 patent, 10:42-60; Gulati, “VMware Distributed Resource Management: Design, Implementation, and Lessons Learned,” *VMware Tech. J.* 1(1), 45-64 (Apr. 2012) (describing DRS for resource allocation); Menascé Decl. ¶¶ 12-13.) The ’752 patent improves conventional resource allocation techniques, teaching “resource allocation analyses for the distributed computer system 200 are not performed by the local resource allocation module, but outsourced to the remote resource allocation module.” (’752 patent, 10:36-60.) In an embodiment, the resource allocation module is on a remote server—aptly designated “remote” in the patent to distinguish it from the resource allocation modules running locally in the distributed computer systems. (*See id.*, 10:42-60, 12:28-35 (“[T]he remote resource allocation module 106 is located in the remove server 104, which can be any type of computer with one or more processors, memory, and other [common computer] components.”), 13:59-14:18; *see also* Menascé Decl. ¶ 14.)

A POSITA understood that the “remote” resource allocation module’s structure is like those conventionally employed as “local” resource allocation modules, such as DRS installed on a vCenter server, but running on a remote computer. (*E.g.*, ’752 patent, 8:7-11 (“In a particular implementation, the local resource allocation module is a distributed resource scheduler (DRS) installed in a VMware vCenter™ server that is executed by one or more processors of the server.”); *see also* Menascé Decl. ¶¶ 12-13.) The ’752 patent confirms this understanding in Figure 6, describing the elements of the exemplary remote resource allocation module. Figure 6

shows “an access interface unit” is part of the remote resource allocation module that “interface[s] with the local resource allocation modules in the different distributed computer systems.” (’752 patent, 12:42-13:14; *see also* Menascé Decl. ¶ 14.) This access interface unit “communicates with the local resource allocation modules” (’752 patent, 12:42-13:14) and receives snapshots from the distributed computer systems (*id.*, 13:15-36), and “results of the resource allocation analysis . . . are then transmitted to the accessing local resource allocation module via the access interface unit” (*id.*, 13:37-58). (Menascé Decl. ¶ 14.) Figure 1 shows in an example that “the distributed computer systems are able to communicate with the remote server [hosting the remote resource allocation module] via the network.” (’752 patent, 3:46-67) The specification teaches that the “network 102 may include the Internet, a wide area network (WAN), a local area network (LAN), a storage area network (SAN), a fibre channel network and/or other networks,” which invoke well-known and standard communication and interface infrastructure for modules on remote servers. (*Id.*, 3:58-63; *see* Menascé Decl. ¶ 14.) The patent teaches a network interface may be a network adapter. (’752 patent, 5:3-4; Menascé Decl. ¶ 14.)

In this context, “remote resource allocation module” evokes sufficient structure and is not subject to § 112(f). That the claims do not recite a particular, commercially-available example of a “remote” resource allocation module, such as DRS installed on a vCenter Server, does not bring the claims under the ambit of § 112(f). *Personalized Media Commc’ns, LLC v. ITC*, 161 F.3d 696, 705 (Fed. Cir. 1998) (“Even though the term ‘detector’ does not specifically evoke a particular structure, it does convey to one knowledgeable in the art a variety of structures known as ‘detectors,’ . . . preclude[ing] the application of § 112, ¶ 6.”)

**The patent satisfies § 112(f)**

Even if the Court concludes that “remote resource allocation module” is a means-plus-function term, the specification and claims recite sufficient structure. The remote resource allocation module refers to a particular disclosed structure that performs the module’s *claimed functions*. See *Creo Prods., Inc. v. Presstek, Inc.*, 305 F.3d 1337, 1344 (Fed. Cir. 2002) (“The function of a means-plus-function limitation . . . must come from the claim language itself.”). Here, the claimed functions are “interfacing with the distributed computer systems,” “receiving a plurality of computer system snapshots from the distributed computer systems” and “transmitting results of the resource allocation analyses . . . to the distributed computer systems.” (’752 patent, claims 1, 9, 17.) Per above, these functions are carried out by an access interface unit of the remote resource allocation module that communicates with the distributed computer systems via standard network interfaces and communication protocols, such as network adapters and Internet protocols. (*Id.*, 3:46-67, 5:3-4, 12:42-13:58.) Thus, if this Court finds “remote resource allocation module” is a means-plus-function claim term, the claim term complies with § 112(f).

#### **b.      Densify’s Responsive Position**

The term “remote resource allocation module” is governed by 35 U.S.C. § 112(f) and is indefinite because the specification does not disclose adequate corresponding structure. A means-plus-function inquiry has two parts. First, the Court must determine whether the claim limitation employs means-plus-function format. Where, as here, a claim does not use the word “means,” there is a rebuttable, not-strong presumption that means-plus-function claiming does not apply. The presumption is overcome, however, by showing by a preponderance of the evidence, that “the claim term fails to recite sufficiently definite structure.” *Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1007 (Fed. Cir. 2018); *Williamson*, 792 F.3d at 1349. The question is “whether the words of the claim are understood by [POSAs] to have a sufficiently definite meaning as the name for structure.” *Zeroclick*, 891 F.3d at 1007. Once a claim is determined to be “means-plus-function,”



the court then identifies the function and assesses whether the specification discloses a corresponding structure. *Williamson*, 792 F.3d at 1351.

Here, means-plus-function applies because the term “remote resource allocation module” is not a name for structure. *Williamson*, 792 F.3d at 1350 (“‘Module’ is a well-known nonce word that can operate as a substitute for ‘means’ .... [It] is simply a generic description for software or hardware that performs a specified function.”); *see also* M.P.E.P. § 2181 (“module,” like “mechanism for,” “device for,” or “system for,” is “a non-structural generic placeholder[]); *TQ Delta, LLC v. 2Wire, Inc.*, 2018 WL 626472, at \*9-10 (D. Del. Jan. 30, 2018) (“message determination module” subject to means-plus-function claiming). Simply appending a function (“remote resource allocation”) to the term “module” does not provide any information as to the relevant structure. The phrase “remote resource allocation module” is not a term of art. Madisetti Dec. ¶¶ 26-31. A POSA may have been familiar with “resource allocation” generally (*see* VMW Op. at 59-60; Menasce Dec. ¶ 11), but the mechanism for conducting it is subjective. ’752 patent at 8:41-44 (“Resource allocation analysis algorithms ... are well known, and thus, are not described herein in detail.”); Madisetti Dec. ¶ 26.

Nothing in the surrounding claim language provides information regarding the structure of “remote resource allocation module.” The claim only describes *where* the module is located (“outside of the distributed computer systems”). Even accepting, *arguendo*, VMware’s assertion that the claims are carried out from the perspective of a computer (or an access interface unit, or the remote resource allocation module itself), the most that could be gleaned from the claim would be the intended *function* of the module, not any information about the underlying structure.<sup>25</sup>

---

<sup>25</sup> VMware’s reliance on *M2M Solutions LLC v. Sierra Wireless America, Inc.*, 2016 WL 1298961 (D. Del. Mar. 31, 2016) is inapposite. The *M2M* Court found that means-plus function claiming did not apply because the claim included additional language that “describes how the

**(i) The Functions “Interfacing ... Receiving ... Performing... and Transmitting”**

The claim is ambiguous as to who or what performs each of the claimed steps, but for purposes of this argument, Densify assumes that the “remote resource allocation module” is responsible for “interfacing ... receiving ... performing... and transmitting.”

VMware agrees that the module is responsible for “interfacing,” “receiving,” and “transmitting,” but not “performing.” D.I. 737-1, Ex. D at 13; VMW Op. at 62. VMware’s position is at odds with the specification, which states that the module is used “for performing resource allocation analyses.” ’752 patent at 13:59-60; *see also id.* at 1:51-55 (a snapshot “is transmitted to a remote resource allocation module so that a remote resource allocation analysis can be performed.”).

**(ii) The Specification Does Not Disclose Corresponding Structure.**

“If a patentee fails to disclose adequate corresponding structure, the claim is indefinite.” *Williamson*, 792 F.3d at 1351-52. The ’752 patent specification does not disclose a sufficient structure for the remote resource allocation module. Fig. 6 purports to describe a remote resource allocation module, but it only names additional components which themselves are devoid of structure: an “access interface unit” and a “remote resource allocation analysis unit.” The specification provides some details as to the operation of the access interface unit, but does not

---

authentication process takes place in considerable detail.” *M2M*, 12-cv-30, D.I. 215 at 8-9 (D. Del. Oct. 2, 2015) (“Despite the fact that the claim recites a function, the immediately following words provide algorithmic structure for performing that function.”). The claim here does not describe how the resource allocation module works. Moreover, *M2M* found that the defendant failed to meet its burden by not submitting any evidence of how a POSA would view the claim term and not addressing the claim language describing how the claimed function was carried out. *Id.* at 8; *see also M2M*, 2016 WL 1298961, at \*6 (“Defendants simply offer conclusory attorney statements....”). In contrast, Densify submits an expert declaration in support of its position and offers an analysis that considers all of the claim language.

provide *any* information regarding the operation or structure of the analysis unit. Instead, the specification states that the analysis unit performs resource allocation analyses “using at least one resource allocation analysis algorithm” (’752 patent, 13:15-21; 13:37-42), but fails to identify or describe any particular algorithm. In fact, the patent explicitly acknowledges its *failure* to provide an algorithm, stating that “[r]esource allocation analysis algorithms that operate on snapshots of distributed computer systems are well known, and thus, are not described herein in detail.” *Id.* at 8:41-44. That is not sufficient under § 112 ¶6. *Arendi*, 2019 WL 3891150 at \*13 (holding “the multiplicity of possible algorithms ... and the specification’s lack of any disclosure of any particular algorithm ..., [causes] the Court [to] conclude[] that the specification fails to disclose sufficient structure for the [relevant] function and, thus, the limitation is indefinite”).

When means-plus-function claims involve the operation of a computer, the specification must explain how the computer actually performs the claimed function. *See, e.g., Aristocrat Techs. Australia Pty Ltd. v. Int’l Game Tech.*, 521 F. 3d 1328, 1333 (Fed. Cir. 2008). This Court’s decision in *Arendi* is informative. In *Arendi*, the Court found four means-plus-function claims indefinite because the specification failed to disclose an algorithm or other description of how to perform the claimed functions. *Arendi*, 2019 WL 3891150 at \*11-15 (“The specification discloses several operations that could be performed ... but does not disclose algorithms to implement these operations.”). Like the plaintiff in *Arendi*, VMware relies on its expert’s opinion “for the proposition that a POSA would have known several ways to [perform resource allocation analyses].” *Id.* at \*13; Menasce Dec. ¶¶ 12-13. But the issue “is not whether a POSA would know, given the specification, how to implement the claimed ... function, but instead whether a POSA would recognize the *specification itself* as disclosing a particular algorithm or algorithms for implementing the ... function.” *Arendi*, 2019 WL 3891150 at \*13. In other words, the claims are

indefinite because “[t]he specification fails to place a POSA on sufficient notice of which [resource allocation analysis] algorithms are claimed – and which are not claimed.” *Id.* at \*15.

**c. VMware’s Reply Position**

Cirba argues that the “‘remote resource allocation module’ is governed by 35 U.S.C. § 112(f) and is indefinite because the specification does not disclose adequate corresponding structure.” (Resp. Br. at 63.) But the parties actually agree that the term “resource allocation module” refers to structures known in the art. (Madisetti Decl. ¶¶ 26-28, 30; *see* Resp. Br. at 63-64.) For this reason alone, the Court should find § 112(f) does not apply to “remote resource allocation module” and give the term its plain and ordinary meaning.

**The parties agree that “resource allocation modules” are well-known structures.**

The parties do not dispute that “resource allocation modules” were structures generally known to POSITAs, including commercially available products. (Menascé Decl. ¶¶ 10-14; Madisetti Decl. ¶¶ 25-28 (“[A] POSA would have been generally familiar with the concept of resource allocation.”), ¶ 30 (“While it is true that VMware’s distributed resource scheduler (DRS) could have been considered a resource allocation module, DRS was far from the only product on the market that performed that function.”); Menascé Rep. Decl. ¶¶ 35-37.) For this reason alone, § 112(f) does not apply. *Chrimar Holding Co., LLC v. ALE USA Inc.*, 732 F. App’x 876, 884-85 (Fed. Cir. 2018), *as amended* (defendant “ha[d]not met its burden to overcome the presumption against applying § 112, ¶ 6 for [alleged nonce words]” because defendant “did not dispute . . . that those terms refer to known structures in the art[;]” “[a] claim term that has an understood meaning in the art as reciting structure is not a nonce word triggering § 112, ¶ 6.”); *Sound View Innovations, LLC v. Facebook, Inc.*, No. 16-116, 2017 WL 2221177, at \*5 (D. Del. May 19, 2017) (“‘Controller’ may be a class of structures, rather than one specific structure, and may be defined with functional terms, but that does not make it means-plus-

function.” (citing *Personalized Media Commc’ns v. ITC*, 161 F.3d 696, 705 (Fed. Cir. 1998))).

Moreover, *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1349 (Fed. Cir. 2015), does not support Cirba’s position that “[s]imply appending a function (‘remote resource allocation’) to the term ‘module’ does not provide any information as to the relevant structure.” (Resp. Br. at 63-64.) The Federal Circuit has clarified that “the mere fact that the disputed limitations incorporate functional language does not automatically convert the words into means for performing such functions.” *Zeroclick, LLC v. Apple Inc.*, 891 F.3d 1003, 1007-09 (Fed. Cir. 2018). A court must consider claim terms in their context. This is particularly true where the context—here, performing resource allocation analyses—“strongly suggests the plain and ordinary meaning.” *See id.* Cirba’s own expert confirmed that the ’752 patent discloses DRS as an exemplary “resource allocation module” among others that were commercially available at the time of invention. (Madisetti Decl. ¶ 30.) *See Chrimar*, 732 F. App’x at 884-85.

**“Resource allocation module” need not refer to one particular structure.**

Instead of disputing that “resource allocation module” refers to known structures, Cirba suggests that the phrase invokes § 112(f) because there are *too many* known “resource allocation modules” from which a POSITA may choose. (See Resp. Br. at 63-64; Madisetti Decl. ¶ 29 (arguing “the multitude of potential implementing structures means that the term ‘remote resource allocation module’ did not have any particular structural meaning”).) Cirba cites no authority for the proposition that the existence of multiple known structures renders a claim term subject to § 112(f) or indefinite. The cases state otherwise. *See, e.g., Personalized Media Commnc’s*, 161 F.3d at 705 (“Even though the term ‘detector’ does not specifically evoke a particular structure, it does convey to one knowledgeable in the art a variety of structures known as ‘detectors.’ We therefore conclude . . . ‘detector’ is a sufficiently definite structural term to

preclude the application of § 112, ¶ 6.”); *Sound View*, 2017 WL 2221177, at \*5 (finding “controller,” a class of functionally-defined structures, is not a mean plus function term).

Cirba’s expert’s opinion supports VMware’s position in view of this case law. He states that “a POSA at the time of the invention could have carried out resource allocation” in accordance with any of four references (Madisetti Decl. ¶ 25) and that DRS, Citrix NetScaler, and Microsoft Virtual Machine Manager Hyper-V were all “resource allocation modules” available for purchase” (*id.* ¶ 30). His opinion confirms that the claim term “resource allocation module” refers to a class of structural components known to those of skill in the art. By invoking such resource allocation modules, the ’752 patent provides a POSITA with sufficient guidance to practice the claims. (Menascé Rep. Decl. ¶ 39.) Cirba therefore cannot overcome the presumption that § 112(f) does not apply.

**The “remote resource allocation module” is not in the “performing” step.**

Even if the Court were to treat this limitation as a means-plus-function limitation, the specification discloses ample supporting structure. (Op. Br. at 62-63.) Cirba bases its argument to the contrary on a supposed failure to disclose a structure and algorithm for the “performing” step. But the “performing” step does not specify that the remote resource allocation module carries out the functions from that step. Cirba contends that this is “at odds with the *specification*” (Resp. Br. at 64-65 (emphasis added)), but the *claims* do not require it. *See Viotech Techs., Inc. v. Microsoft Corp.*, C.A. No. 14-1226-RGA, 2016 WL 3398025, at \*10 (D. Del. Jun. 14, 2016) (“The identified function must be the function ‘explicitly recited in the claim.’” (quoting *Micro Chem, Inc. v. Great Plains Chem. Co.*, 194 F.3d 1250, 1258 (Fed. Cir. 1999))). “Remote resource allocation module” appears nowhere in the “performing” step of claims 1 and 9.

**d.      Densify’s Sur-Reply Position**

35 U.S.C § 112(f) applies because “remote resource allocation module” is a nonce term associated with the “interfacing,” “receiving,” “performing,” and “transmitting” functions.<sup>26</sup> It does not connote any particular structure, and the specification provides none. VMware’s reply misstates Densify’s position and misapplies the law. The fact that “resource allocation modules” were known does not mean that “*remote* resource allocation modules” were known, a condition VMware’s expert, Dr. Menasce, acknowledges.<sup>27</sup> Menasce Dec. II ¶ 37 (“[A]t the time of the invention, *remote* resource allocation ... was *not* well known to POSAs”) (emphasis added).

The specification does not provide any structural information about the RRAM or the algorithm used to perform remote resource allocation. The Court’s decision in *Arendi* explains why that renders the claim indefinite. *Arendi*, 2019 WL 3891150, \*11-15 (D. Del. Aug. 8, 2019). VMware argues that “the ’752 patent provides a POSA with sufficient guidance to practice the claims,” but this Court rejected the same argument from the same expert. *Id.* at \*13 (“The issue ... is not whether a POSA would know...how to implement the ... function, but instead whether a POSA would recognize the specification itself as disclosing a particular algorithm ....”).<sup>28</sup>

---

<sup>26</sup> VMware argues that the RRAM is *not* responsible for carrying out the “performing” step but offers no argument as to what is. Because the claim itself does not recite what does the “performing” step, it must be interpreted in light of the specification, which makes clear that it is the RRAM. ’752 patent at 13:59-60; *see also id.* at 1:51-55 (a snapshot “is transmitted to a remote resource allocation module so that a remote resource allocation analysis can be performed”).

<sup>27</sup> Even if “resource allocation modules” could somehow stand-in for “remote resource allocation modules,” then, as VMware acknowledges, any structure facilitating resource allocation could fairly be characterized a “resource allocation module.” Reply at 16-17. A “claim ... cannot be construed so broadly to cover every conceivable way or means to perform the function....” *Mas-Hamilton Grp. v. LaGard, Inc.*, 156 F.3d 1206, 1214 (Fed. Cir. 1998).

<sup>28</sup> The fact that the patent identifies vCenter’s “distributed resource scheduler” (DRS) as a “local resource allocation module” does not means the patent discloses sufficient structure for a *remote* resource allocation module. But even if it did, the patent does not identify what algorithm DRS should employ to perform its resource allocation. Madisetti Dec. ¶ 25.

### 3. “receiving a plurality of computer system snapshots” (Claims 1, 9)

VMware’s Proposed Construction	Densify’s Proposed Construction
This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

#### a. VMware’s Opening Position

This step is not indefinite. It uses ordinary, clear English to specify the data—“a plurality of computer system snapshots”—received at the remote resource allocation module. As long as two or more computer system snapshots are received, the limitation is satisfied. The Court therefore should decline to construe this term.

The specification confirms that the ordinary language of these claims applies. For example, the specification teaches that in an embodiment the remote resource allocation module receives “a snapshot of the distributed computer system . . . from the local resource allocation module.” (’752 patent, 13:15-21.) It also teaches an exemplary method that “obtain[s] a snapshot of a target distributed computer system at the distributed computer system” and “transmit[s] the snapshot of the target distributed computer system to the remote resource allocation module,” where it is received. (*Id.*, 1:58-2:13; *see also id.*, 14:41-61 (same).)

#### b. Densify’s Responsive Position

This term is indefinite because it permits two constructions of differing scopes, and the specification fails to inform a POSA, with reasonable certainty, which is the proper one.

The portions of the claims at issue are:

A method for performing remote resource allocation analyses on *a plurality of distributed computer systems*, the method comprising:

\* \* \*

receiving *a plurality of computer system snapshots* from *the distributed computer systems* at the remote resource allocation module;

The preamble, which the parties agree is limiting (D.I. 737-1, Ex. D at 12), recites a “plurality of distributed computer systems” which is then referenced as “the” distributed computer systems in



the “receiving” element. The “receiving” step requires that a “plurality of computer system snapshots” be received from these distributed computer systems, but does not make clear whether “a plurality of . . . snapshots” must come from *each* of the distributed computer systems, or if they may be received from the *totality* of the distributed computer systems.<sup>29</sup> Neither the claim nor the specification provides a POSA the means to ascertain which construction should apply. *See Red Rock Analytics, LLC v. Samsung Elecs. Co., Ltd.*, 2018 WL 1806859, at \*19 (E.D. Tex. Apr. 16, 2018) (claim indefinite because it was “unclear as to whether the [term] has a meaning that is singular, plural, or both”); Madisetti Dec. ¶¶ 32-37. In fact, the specification lacks a written description for either construction. Thus, even if a POSA were able to choose between the two constructions, the POSA would still be unable to practice the claimed invention.

**(i) First possible construction – receiving a plurality of snapshots from *each* of the plurality of distributed computer systems**

The specification consistently describes the invention in terms of individual distributed computer systems. *See, e.g.*, ’752 patent at 1:58-2:10 (describing each claim limitation through reference to “*a* distributed computer system” (emphasis added), 14:41-61; Fig. 7; 9:40-45 (describing “interfacing” step through reference to a single DCS), 13:15-42 (for “receiving” and “performing” steps); 13:53-58 (for “transmitting” step). The specification includes an example of Y-number DCSs (“[t]he distributed computer systems F-1, F-2 ...F-Y are facilities with a collection of computers and storage devices”) and describes the claim steps with respect to each. *Id.* at 12:44-47 (explaining that the remote resource allocation module interfaces with “the different distributed

---

<sup>29</sup> A simple hypothetical makes the ambiguity clear: if the plurality of snapshots totals 2 snapshots, and the plurality of distributed computer systems (“DCSs”) totals 3 systems, does the “receiving” step require 2 snapshots be received from *each* of the DCS (making a total of 6 snapshots) *or* is it sufficient to receive 2 snapshots from *all* 3 of the DCSs (making a total of 2 snapshots)?

computer systems F-1, F-2 ... F-Y”); 12:61-64 (remote resource allocation module can “contain subscription agreement information for *each* of the different local resource allocation modules”) (emphasis added). Thus, it is conceivable for a POSA to conclude that a “plurality of ... snapshots” must be received from *each* DCS in the plurality of claimed DCSs.

This construction, however, would require “performing remote resource allocation analyses” on the “plurality of ... snapshots” from each DCS (’752 patent, cl. 1 (the “performing” element follows the “receiving” element)), but the specification does not provide any description as to how to do that. The specification only describes resource allocation analysis with respect to a *single* snapshot. *See, e.g.*, Fig. 7 (one obtains “*a* snapshot” of a distributed computer system and “[t]ransmit[s] *the* snapshot” to a remote resource allocation module (emphases added)); 8:14-17; 10:6-11 (“[T]he remote processing interface unit 406 transmits the snapshot ... so that a resource allocation analysis can be performed on the snapshot[.]”); 13:37-42 (“The remote resource allocation analysis unit 604 operates to process the received snapshot ... .”); 14:53-58. In fact, the specification *never* refers to multiple snapshots when describing remote resource allocation—every disclosed embodiment involves the use of only a *single* snapshot per DCS. *See* 1:58-2:10 (“A method ... in accordance with an embodiment of this invention comprises obtaining a snapshot ... transmitting the snapshot ... and receiving ... results of the resource allocation analysis on the snapshot”); Fig. 7; 8:26-38 (describing three possible “modes of operation,” each involving transmitting and processing a single snapshot).

The difference between performing remote resource allocation analysis on a single snapshot vs. multiple snapshots “is not trivial.” Madisetti Dec. ¶¶ 34-37. Multi-snapshot analysis involves snapshots containing different data. For example, two snapshots can differ with respect to (a) when they were taken, (b) the system components included in the snapshot, (c) the level of

granularity at which the snapshots were taken, (d) the data or metrics reflected in the snapshot, (e) the tools used to collect the data, or (f) any combination of the above. *Id.* The specification does not address how multiple snapshots are collected or how resource allocation analyses is performed on multiple snapshots from the same DCS. *Id.*

**(ii) Second possible construction – receiving a plurality of snapshots from the *totality* of the plurality of distributed computer systems**

The claim language also allows for the possibility that two or more (“a plurality of”) snapshots are received from *all* (not each) of the plurality distributed computer systems. A POSA may conclude that this construction is the right one because, as discussed above, the alternative “each of” interpretation excludes every disclosed embodiments. *Kaneka Corp. v. Xiamen Kingdomway Grp. Co.*, 790 F.3d 1298, 1304 (Fed. Cir. 2015) (“A claim construction that excludes a preferred embodiment is rarely, if ever, correct. A construction that excludes *all* disclosed embodiments ... is especially disfavored.” (quotation marks and citation omitted)); *CSS Tech.*, 778 F. App’x at 950. This construction allows for the possibility that each DCS sends only a single snapshot.<sup>30</sup> But the “totality” construction faces its own indefiniteness problem. Under this construction, remote resource allocation analyses must be performed on “[the] plurality of [DCS]s” even if a given DCS does not send any snapshots.<sup>31</sup> The specification provides no disclosure covering this scenario – *every* embodiment in the specification requires a single snapshot from each DCS, and no other possibilities are so much as alluded to.

---

<sup>30</sup> The “each of” construction would *never* permit a DCS to send only a single snapshot because the limitation expressly requires a *plurality* of snapshots.

<sup>31</sup> In the hypothetical from Footnote 29, two snapshots may be received, for example, from the group of three DCSs, such that two single snapshots came from two of the three DCS, but the third DCS sends none.

**c. VMware's Reply Position**

Cirba argues that a POSITA could not determine whether the received “plurality of computer system snapshots” “must come from *each* of the distributed computer systems, or if they may be received from the *totality* of the distributed computer systems.” (Resp. Br. at 71-74 (emphasis original).) Cirba’s argument has no basis in the claim language, which merely requires “receiving a plurality of computer system snapshots” from the “plurality of distributed computer systems.” (Op. Br. at 70-71.)

Per Cirba, the POSITA’s alleged confusion flows from an inability to determine whether an “each of” interpretation applies to the claims. (Resp. Br. at 72-73.) But the phrase “each of” appears nowhere in this claim limitation. (Menascé Rep. Decl. ¶ 42.) And Cirba offers no reason why the Court should entertain Cirba’s indefiniteness argument based on adding hypothetical “each of” language to the independent claims.

Cirba also asserts that its “each of” interpretation excludes every disclosed embodiments [sic],” creating a “problem.” (Resp. Br. at 72-73.) But its own authority, *Kaneka Corp. v. Xiamen Kingdomway Group Co.*, solves the problem. An interpretation that excludes all embodiments does not render the claim indefinite, as the court should not adopt it. *See Kaneka*, 790 F.3d 1298, 1304-05 (Fed. Cir. 2015) (construing term consistently with intrinsic record and noting a construction that “excludes *all* disclosed embodiments . . . is especially disfavored”).

Cirba’s “indefiniteness” argument regarding its “totality” interpretation fares no better. Cirba argues that “the specification provides no disclosure covering [the] scenario [where not every distributed computer system among the plurality sends a snapshot]” (Resp. Br. at 73-74), but that is not a reason to find a claim indefinite. Nothing in the claims requires that *all* distributed computer systems *must* send a snapshot. Moreover, determining the appropriate plurality of computer system snapshots to receive from the plurality of distributed computer

systems for a given use case, including a scenario where not every distributed computer system sends a snapshot, is well within the skill of a POSITA. (Menascé Rep. Decl. ¶ 43.)

**d. Densify’s Sur-Reply Position**

This term is indefinite because it can be construed two ways (either a plurality of snapshots from *each* DCS, or a plurality of snapshots from *the totality* of the DCSs) and because either construction lacks written description.<sup>32</sup> VMware does not dispute that the specification does not explain how one could perform resource allocation analysis on a DCS *without* receiving a snapshot from the DCS, thereby making the “totality of” construction indefinite for lack of written description. And because VMware acknowledges that the “each of” construction is also improper (VMW Reply at 74), there is no reasonable way to read this claim that does not render it indefinite.

**4. “as defined by the computer system snapshots” (Claims 3, 11)**

<b>VMware’s Proposed Construction</b>	<b>Densify’s Proposed Construction</b>
This term should not be construed or should be given its plain and ordinary meaning.	Indefinite

**a. VMware’s Opening Position**

Cirba selected this phrase for construction from a larger limitation in Claims 3 and 11: “wherein performing the remote resource allocation analyses includes generating one or more recommendations regarding resource allocations for the distributed computer systems as defined by the computer system snapshots.” This limitation adds a “generating step to the “performing” step in claim 1. In this context, the phrase “recommendations regarding resource allocations for the distributed computer systems as defined by the computer system snapshots” merely refers to

---

<sup>32</sup> VMware argues that “each of” does not appear in the claim. That is true. Neither does “totality of.” But simply as a matter of language, one construction or the other must apply – there is no other reasonable way to interpret the claim.

recommendations for resource allocations resulting from the analyses in the “performing” step, with no need for further construction. It is not indefinite.

The specification supports this understanding. It explains that “[t]he results of the resource allocation analysis may also include a recommendation to change the resource entitlement for one or more clients or host computers in the distributed computer system 200 based at least on the current usage of a particular resource, as defined in the received snapshot.” (’752 patent, 8:64-9:2; *see also id.*, 13:37-42.)

#### **b.      Densify’s Responsive Position**

This term is indefinite because it is not clear *what* is “defined by the computer system snapshots.” The claim contains two possible referents: “one or more recommendations regarding resource allocations,” and “the distributed computer systems.” ’752 patent, cls. 3, 11. But neither makes sense. On one hand, to say that the *recommendations* are defined by the computer system snapshots is nonsensical because snapshots do not contain recommendations—they simply describe “configurations and resource usage information.” D.I. 737-1, Ex. D at 12. But to say that the snapshots define *the distributed computer systems* is also nonsensical because the specification makes clear that the snapshots need not correspond to or describe an actual distributed computer systems; instead, snapshots “can be viewed as a snapshot of an imaginary or hypothetical distributed computer system.” ’752 patent at 11:2-6. For example, suppose a DCS has three Windows-based hosts and sends a snapshot to the remote resource allocation module depicting a system that has 50 Linux-based hosts. In that scenario, it would not make sense to say that the snapshot *defined* the DCS, since the system described by the snapshot did not bear any *resemblance* to the DCS. Because there is no way to discern what, if anything, in the claim is “defined by the computer system snapshots,” a POSA would have no way of knowing whether any generated recommendations infringe this term.

**c. VMware's Reply Position**

Ignoring plain meaning (Op. Br. at 76), Cirba offers dueling interpretations that are both plainly at odds with the claim language and the specification, (Resp. Br. at 30). The patent teaches that the *computer system* snapshots define the states of the distributed *computer systems* under analysis—*e.g.*, their current usage of a particular resource. ('752 patent at 7:39-62; Menascé Rep. Decl. ¶ 44; Op. Br. at 76.) Cirba even highlights this point, agreeing that the snapshots “describe ‘configurations and resource usage information.’” (Resp. Br. at 76-77.) A POSITA would understand that the methods of claims 1 and 3 analyze configurations and resource usage information of the distributed computer systems. (Menascé Rep. Decl. ¶ 44.)

**d. Densify's Sur-Reply Position**

VMware's Reply does not address Densify's arguments. One cannot say the snapshots define the DCSs, since snapshots can contain imaginary or hypothetical data with no connection to any of the underlying systems. '752 patent, 11:2-6.

Dated: July 9, 2020

*/s/ Anne Shea Gaza*

---

Anne Shea Gaza (No. 4093)  
Robert M. Vrana (No. 5666)  
Samantha G. Wilson (No. 5816)  
Rodney Square  
1000 North King Street  
Wilmington, DE 19801  
(302) 571-6600  
agaza@ycst.com  
rvrana@ycst.com  
swilson@ycst.com

*Attorneys for VMware, Inc.*

*/s/ Kenneth L. Dorsney*

---

Kenneth L. Dorsney (#3726)  
kdorsney@morrisjames.com  
MORRIS JAMES LLP  
500 Delaware Avenue, Suite 1500  
Wilmington, DE 19801  
Telephone: (302) 888-6800

*Attorneys for Cirba Inc. (d/b/a DENSIFY)  
and Cirba IP, Inc.*